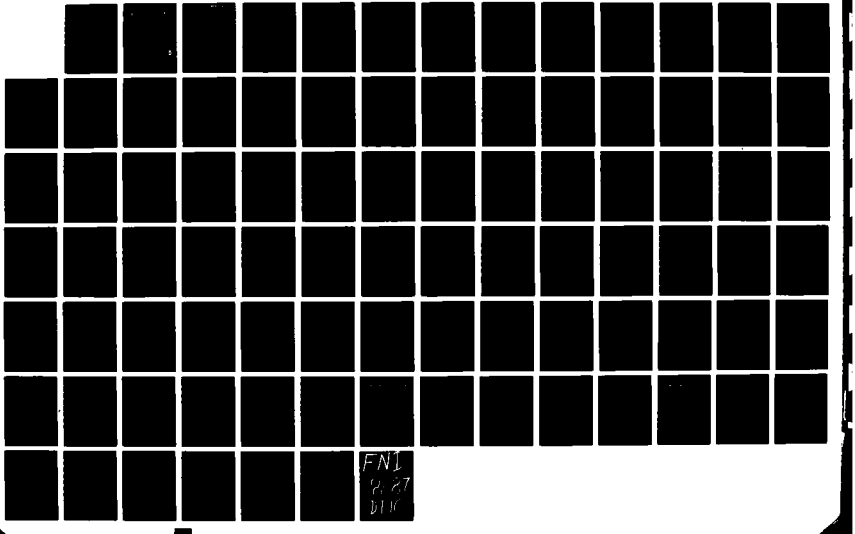
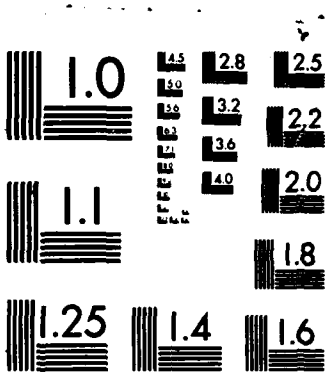
AD-A182 424 TSAR (THEATER SIMULATION OF AIRBASE RESOURCES) OUTPUT PROCESSOR DOCUMENTATION(U) ORLANDO TECHNOLOGY INC SHALIMAR FL R LAMARCHE ET AL. 29 MAY 87 1/1

UNCLASSIFIED F/G 15/5 NL





(2)

Air Force Center for Studies & Analyses

AFISA TO SHED LIGHT

TSAR

OUTPUT PROCESSOR DOCUMENTATION

29 MAY 1987

Prepared for
TACTICAL SUPPORT DIVISION
Air Force Center for Studies and Analyses
Pentagon, Washington, D.C. 20330-5420

Prepared by
ORLANDO TECHNOLOGY, INC.
P.O. Box 855
Shalimar, Florida 32579

87

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A182 424

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Unlimited Distribution
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Orlando Technology, Inc. 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Tactical Support Division Air Force Center for Studies and Analyses
6c. ADDRESS (City, State and ZIP Code) P.O. Box 855 Shalimar, Florida 32579	7b. ADDRESS (City, State and ZIP Code) AFCSA/SAGP Pentagon, Rm 1C370 Washington DC 20330-5420
8a. NAME OF FUNDING/SPONSORING ORGANIZATION AF Center for Studies and Analyses 8b. OFFICE SYMBOL (If applicable) AFCSA/SAGP	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code) AFCSA/SAGP Pentagon, 1C370 Washington DC 20330-5420	10. SOURCE OF FUNDING NOS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO
11. TITLE (Include Security Classification) TSAR F-4E, F-16, F-15, TSAR Output Processor Documentation	
12. PERSONAL AUTHOR(S) Robert Lamarche, Cliff Gornto	
13a. TYPE OF REPORT Final 13b. TIME COVERED FROM 8/1/85 TO 7/31/86	14. DATE OF REPORT (Yr., Mo., Day) 870417 15. PAGE COUNT
16. SUPPLEMENTARY NOTATION	

17 COSATI CODES		
FIELD	GROUP	SUB GR

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
TSAR, Simulation, Sortie Generation, Logistics, Capability Assessment, Support, Graphics.

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A significant problem for analysts and simulation model users is presenting output data in convenient form. The Theater Simulation of Airbase Resources (TSAR) model is no exception. This TSAR document provides users with options for generating data outputs in a format which can be directly input into standardized analytic software packages. TSAR implementation code is listed and sample analytic and graphic procedures illustrate the use of various options.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL LtCol Salvatore R. Bosco	22b. TELEPHONE NUMBER (Include Area Code) (202) 697-0862 22c. OFFICE SYMBOL AFCSA/SAGP

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

PREFACE

The Assistant Chief of Staff for Studies and Analyses (AF/SA) has a continuing requirement for investigations into advanced fighter aircraft operations and support topics. A recurring need involves studies of readiness, survivability, and sustainability. Several methodologies have been used over the years. The current state-of-the-art techniques for these purposes are two Monte Carlo simulation models developed in the late 1970s by The Rand Corporation, Theater Simulation of Airbase Resources (TSAR) and TSAR Inputs using Airbase Damage Assessment (TSARINA). These models, like other simulation models, are built to study and analyze a system's processes. In this case the "system" is the collection of resources called an airbase and the process of interest is the interaction of those resources resulting in the generation of aircraft sorties. A system's problem can often be described and studied through "what if" excursions about a defined base case. The base case and the excursions of interest could be viewed as related problem scenarios. In both TSAR and TSARINA the scenario to be studied is modeled through the database. Therefore the analyst must know the logic embodied in the program structure, but most importantly, completely understand the scenario as described in an extensive database. The differences between scenarios involving the same aircraft type may only involve changing several cards, but building the components of the baseline database and/or acquiring sufficient understanding of what is contained in such a database are significant tasks. Hence the need for a disciplined development and adequate documentation. Given that a baseline database exists, the modeler must replace, merge, or modify various database segments to fashion a new scenario or to specify excursions from the base case. Alternative data segments which are clearly documented are therefore often needed. The availability and limited documentation of databases for both TSAR and TSARINA impose practical limitations to their usefulness.

The author of TSAR and TSARINA, Don Emerson, has provided analysts with extremely powerful tools for tactical support analysis. They are very well written and documented. The real problem for the analyst is locating sources of data to make use of the full richness inherent in the models. It was clear to those of us at the Air Force Center for Studies and Analyses (AFCSA) that if our results and observations were going to be credible, the databases and assumptions they embodied would need to be documented. Our intent was to collect selected databases within AFCSA to support current and projected studies. Quality documentation of these databases was necessary to permit analysts to understand the assumptions, limitations, and level of detail that was being portrayed. The resultant availability of databases and standardization of documentation will not only directly support in-house investigations but will also facilitate studies across the analysis community. Because of the scope of such a task, a contract was let to ensure its timely accomplishment.

Orlando Technology, Inc., was awarded a competitive contract for TSAR/TSARINA support tasks. The tasks focus around the model databases and database segments. They began with the existing model databases and updated them based on the most current government data available. These databases were to be documented in three ways. The first is a dictionary for each database and separate database segment, which translates the database codes to their English equivalents. Secondly, graphic network models are needed for those portions of the database which model decision logic networks for repair tasks. And finally,

an index is needed to cross-reference the database segments, dictionary, and the network models to facilitate their use by modelers and analysts. The long term intent is to build on these basic databases by the use of a database management system to facilitate changes, updates, and analysis scenario development. As the models mature and the user community grows, the model databases will continue to evolve and grow in depth and breadth. This document should be viewed as an advanced prototype which will hopefully continue to mature and increase in usefulness.

TSAR outputs can be formidable and comparisons between TSAR runs may involve collating large printouts; this document was developed to facilitate the use of TSAR output. The output processor produces specific TSAR results in a format compatible with standardized analytic software packages and is a user controlled option. Users assign a 0 or 1 to variables identified on a new TSAR card, 2-7, to toggle outputs on or off. The resulting TSAR output can then be input into the analytic packages to produce the desired product either on tabular or graphic form.

Our hope is that you will wear out this document through constant usage. Pass along your comments and criticisms so that future improvements can incorporate the user community's collective insights.

SALVATORE R. BOSCO, Lt Col, USAF
AFCSA/SAGP
Washington, DC 20330-5420

TSAR OUTPUT PROCESSOR DOCUMENTATION

May 1987

PREPARED FOR

AIR FORCE CENTER FOR STUDIES AND ANALYSES
PENTAGON, WASHINGTON D.C. 20330-5420

PREPARED BY

ORLANDO TECHNOLOGY, INCORPORATED
P.O. BOX 855
SHALIMAR, FLORIDA 32579

CONTENTS

I. TSAR OUTPUT PROCESSOR

The main purpose of the output processor is to better facilitate the analysis of TSAR results. The output processor produces specific TSAR results in a format compatible with standard analytic sorftware packages and is a user controllable option. The multiple TSAR results available include daily summaries of sorties flown and demanded, aircraft damaged and lost, cannibalizations, NMCS hours and aircraft, on- and off-equipment delays due to limitations of personnel and AGE, delays due to spare parts, and quantities and times of on-equipment tasks and off-equipment repairs.

In this documentation, we used the SAS® System as our analysis output software. Other packages may be used; however, the TSAR output may need to be modified to make it compatible with package requirements. Our use of SAS in no way implies that we endorse the product.

A. Input/Out

The following table lists the FORTRAN variable name and the logical unit number associated with each output processor data type. The logical unit number 81 translates into output file name "FOR081" or "FOR081.DAT".

Data Type	Variable Name	Logical Unit Number
Daily Sorties Flown and Demanded	SAS1	81
Aircraft Lost	SAS2	82
Aircraft Damaged	SAS3	83
Cannibalizations	SAS4	84
NMCS Hours	SAS5	85
NMCS Aircraft	SAS6	86
On-Equipment delays due to limited personnel	SAS7	87
Off-Equipment delays due to limited personnel	SAS8	88
On-Equipment delays due to limited AGE	SAS9	89
Off-Equipment delays due to limited AGE	SAS10	90

SAS is a registered trademark of SAS Institute Inc., Carey, NC USA.

Delays due to spare parts	SAS11	91
Quantities/Times of On-Equipment Tasks	SAS12	92
Quantities/Times of Off-Equipment Repairs	SAS13	93

Since the output processor is to be controlled by the user, a new TSAR input card type has been created. On this input card, for each variable set equal to 1, the corresponding output processor data type will be written to the corresponding logical unit or output file in a format compatible with SAS. This card type is the 2/7 card and follows the standard TSAR input format as described below:

Columns	Variable	Value	Explanation
1-2	I	2	TSAR Card type
3-5	J	7	TSAR Card subtype
6-10	SAS1	0 or 1	SAS1 Flag
11-15	SAS2	0 or 1	SAS2 Flag
16-20	SAS3	0 or 1	SAS3 Flag
21-25	SAS4	0 or 1	SAS4 Flag
26-30	SAS5	0 or 1	SAS5 Flag
31-35	SAS6	0 or 1	SAS6 Flag
36-40	SAS7	0 or 1	SAS7 Flag
41-45	SAS8	0 or 1	SAS8 Flag
46-50	SAS9	0 or 1	SAS9 Flag
50-55	SAS10	0 or 1	SAS10 Flag
56-60	SAS11	0 or 1	SAS11 Flag
61-65	SAS12	0 or 1	SAS12 Flag
66-70	SAS13	0 or 1	SAS13 Flag

B. TSAR Modifications

The following paragraphs describe the required modifications for each data type and the TSAR subroutines, variables and data arrays affected. Appendix A contains the TSAR implementation code.

1. Common Block

FORTRAN labeled common block, "SAS", contains all of the output processor variables and is included in those subroutines that contain the output processor code. These arrays are dimensioned for the maximum values in TSAR. The amount of memory required for these arrays may be reduced by redimensioning the arrays to the maximum number of TSAR assets the user expects to encounter. That is, if the database being processed had only 1000 parts, 2 aircraft and 2 bases, SASCAN and SASPART could be redimensioned from (3199, 9, 63) to (1000,2,2).

2. Daily Sorties Flown and Demanded

The number of sorties flown and demanded each day is printed by subroutine OUTPUT. This subroutine is called at the end of each day during the simulation. The cumulative number of sorties flown and demanded by mission priority, mission type, aircraft type and base number is stored in data array OUTPT1(2, PRTY, MAXM, MAXT, MAXB) and OUTPT1(1, PRTY, MAXM, MAXT, MAXB) respectively.

Subroutine OUTPUT tests SAS1. If SAS1 is equal to 1, a record will be written to logical unit 81. This record contains the trial number, day, base, aircraft type, mission type, mission priority, number of sorties flown and number of sorties demanded.

3. Aircraft Lost

The cumulative number of aircraft lost is stored in the variable LOSTAC which is the number of all aircraft types lost. To distinguish the number of aircraft lost by type, a new data array (SASLST) was created. SASLST is dimensioned according to the maximum number of aircraft types (MAXT) and number of bases (MAXB) simulated in TSAR. Subroutine KILLAC updates this array each time an aircraft is lost. Subroutine OUTPUT tests SAS2. If SAS2 is equal to 1, a record is written to logical unit 82. This record contains the trial number, day, base, aircraft type and quantity of aircraft lost.

4. Aircraft Damaged

The cumulative number of aircraft damaged is stored in the variable DAMAC which is the number of all aircraft types damaged. To distinguish the number of aircraft damaged by type, a new data array (SASDAM) was created. SASDAM is dimensioned according to

the maximum number of aircraft types (MAXT) and number of bases (MAXB) simulated in TSAR. Subroutine PSTFLT updates this array each time an aircraft is damaged. Subroutine OUTPUT tests SAS3. If SAS3 is equal to 1, a record is written to logical unit 83. This record contains the trial number, day, base, aircraft type and quantity of aircraft damaged.

5. Cannibalizations

The cumulative number of cannibalizations by base is stored in BASES(6,BASE) and is updated in subroutine CANNIB. To distinguish the number of cannibalizations by part number, aircraft type and base, a new data array (SASCAN) was created. SASCAN is dimensioned according to the maximum number of parts (NPART), the number of aircraft types (MAXT) and number of bases (MAXB). Subroutine CANNIB updates this array each time a part is cannibalized. Subroutine OUTPUT tests SAS4. If SAS4 is equal to 1, a record is written to logical unit 84. This record contains the trial number, day, base, part number and number of cannibalizations.

6. NMCS Hours

The cumulative number of NMCS hours is stored in data array NORHRS(BASE) and is the NMCS hours for all aircraft types for a given base. To distinguish the NMCS hours by aircraft type and base, a new data array (SASNMH) was created. SASNMH is dimensioned according to the maximum number of aircraft types (MAXT) and the maximum number of a bases (MAXB) simulated in TSAR. Subroutine MANAGE was modified to update this array. Subroutine OUTPUT tests SAS5. If SAS5 is equal to 1, a record is written to logical unit 85. This record contains the trial number, day, base, aircraft type and the NMCS hours.

7. NMCS Aircraft

The cumulative number of NMCS aircraft is stored in data array NOR(BASE) which is the number of NMCS aircraft for all aircraft types for a given base. To distinguish the NMCS aircraft by aircraft type and base, a new data array (SASNMA) was created. SASNMA is dimensioned according to the maximum number of aircraft types (MAXT) and the maximum number of bases (MAXB) simulated in TSAR. Subroutine MANAGE updates this array. Subroutine OUTPUT tests SAS6. If SAS6 is equal to 1, a record is written to logical unit 86. This record contains the trial number, day, base, aircraft type and the number of NMCS aircraft.

8. On- and Off-Equipment Delays Due to Personnel

Delays due to limitations of personnel are stored in OUTPER(2,NOPEOP,MAXB) for on-equipment tasks and in OUTPEO(2,NOPEOP,MAXB) for off-equipment repairs. To distinguish these delays by personnel type and base, two new data arrays (SASPER and SASPEO) were created. SASPER and SASPEO are

dimensioned according to the maximum type of personnel (NOPEOP) and maximum number of bases (MAXB) simulated in TSAR. To distinguish the number of on- and off-equipment delays for each personnel type, two new variables (CNTPER and CNTPEO) were created. SASPER, SASPEO, CNTPER and CNTPEO are updated in subroutine TIMES. Subroutine OUTPUT tests SAS7 and SAS8. If SAS7 is equal to 1, a record consisting of trial number, day, base, personnel type, on-equipment delay time, and number of delays is written to logical unit 87. If SAS8 is equal to 1, a record consisting of trial number, day, base, personnel type, off-equipment delay time and the number of delays is written to logical unit 88.

9. On- and Off-Equipment Delays Due to AGE

Delays due to limitations of AGE are stored in OUTAGE(2,NOAGE,MAXB) for on-equipment tasks and OUTEQP(2,NOAGE,MAXB) for off-equipment repairs. To distinguish these delays by AGE type and base, two new data arrays (SASAGE and SASEQP) were created. SASAGE and SASEQP are dimensioned according to the maximum type of AGE (NOAGE) and maximum number of bases (MAXB) simulated in TSAR. To distinguish the number of on- and off-equipment delays for each AGE type, two new variables (CNTAGE and CNTEQP) were created. SASAGE, SASEQP, CNTAGE and CNTEQP are updated in subroutine TIMES. Subroutine OUTPUT tests SAS9 and SAS10. If SAS9 is equal to 1, a record consisting of trial number, day, base, AGE type, on-equipment delay time and number of delays is written to logical unit 89. If SAS10 is equal to 1, a record consisting of trial number, day, base, AGE type, off-equipment delay time and the number of delays is written to logical unit 90.

10. Delays Due to Spare Parts

Delays due to limitation of spare parts is stored in OUTPRT(2,NOPART,MAXB). To distinguish this delay by spare part type and base, a new data array (SASPRT) was created. SASPRT is dimensioned according to the maximum number of spare parts (NOPART) and the maximum number of bases (MAXB) simulated in TSAR. To distinguish the number of delays for each spare part, a new variable (CNTPRT) was created. SASPRT and CNTPRT are updated in subroutine TIMES. Subroutine OUTPUT tests SAS11. If SAS11 is equal to 1, a record consisting of trial number, day, base, spare part type, delay time and number of delays is written to logical unit 91.

11. Quantities/Times of On-Equipment Tasks

Quantities and times of on-equipment tasks are stored in OUTSHP(2,SHOP,MAXB). To distinguish the task times by shop and base, a new data array (SASON) was created. SASON is dimensioned according to the maximum number of shops (SHOP) and the maximum number of bases (MAXB) simulated in TSAR. To distinguish the number of times a shop performs an on-equipment task, a new

variable (CNTON) was created. SASON and CNTON are updated in subroutine TIMES. Subroutine OUTPUT tests SAS12. If SAS12 is equal to 1, a record consisting of trial number, day, base, shop, task time, and number of tasks is written to logical unit 92.

12. Quantities/Times of Off-Equipment Repairs

Quantities and times of off-equipment repairs are stored in OUTSHP(5,SHOP,MAXB). To distinguish the repair times by shop and base, a new data array (SASOFF) was created. SASOFF is dimensioned according to the maximum number of shops (SHOP) and the maximum number of bases (MAXB) simulated in TSAR. To distinguish the number of times a shop performs an off- equipment task, a new variable (CNTOFF) was created. SASOFF and CNTOFF are updated in subroutine TIMES. Subroutine OUTPUT tests SAS13. If SAS13 is equal to 1, a record consisting of trial number, day, base, shop, repair time, and number of repairs is written to logical unit 93.

13. TSAR Output Summaries

At the end of each day, subroutine OUTPUT writes a record consisting of the total number of aircraft damaged, aircraft lost, NMCS aircraft hours and number of NMCS aircraft to logical unit 6. To accumulate the number of aircraft damaged, aircraft lost, NMCS aircraft hours and number of NMCS aircraft for each trial, four new data arrays (SASDAMT, SASLSTT, SASNMHT, and SASNMAT) were created. At the end of each day, subroutine OUTPUT updates SASDAMT,SASLSTT,SASNMHT, and SASNMAT. At the end of each trial, subroutine SUMUP writes a record consisting of SASDAMT,SASLSTT,SASNMHT, and SASNMAT to logical unit 6.

C. Summary of Subroutine Updates

MANAGE

Accumulates the number of NMCS aircraft hours for each base by aircraft type (SASNMH).

INIT

Initializes the value of SASLST, SASLSTT, SASDAM, SASDAMT, SASNMA, SASNMAT, SASNMH, SASNMHT, SASPER, SASPEO, CNTPER, CNTPEO, SASAGE, CNTAGE, CNTEQP, SASEQP, CNTPRT, SASPRT, SASCAN, SASON, SASOFF, CNTON and CNTOFF to zero after each trial.

INPUT

Reads the 2/7 card and checks the value of SAS1, SAS2, SAS3, SAS4, SAS5, SAS6, SAS7, SAS8, SAS9, SAS10, SAS11 SAS12 and SAS13. Writes the header to the appropriate logical unit.

OUTPUT

Writes the data to the appropriate logical unit after each day and accumulates the total of number of aircraft damaged, aircraft lost, NMCS aircraft hours and number of NMCS aircraft for each trial. Converts TTU's to hours for variables SASPER, SASPEO, SASAGE, SASEQP, SASPRT, SASON, and SASOFF.

Writes a daily summary of the number of aircraft damaged, aircraft lost, NMCS aircraft hours and number of NMCS aircraft to logical unit 6. Initializes variables SASLST, SASDAM, SASNMH, and SASNMA to zero after daily summary is written.

Accumulates the total number of aircraft damaged (SASDAMT), aircraft lost (SASLSTT), NMCS aircraft hours (SASNMHT), and number of NMCS aircraft (SASNMAT) for each trial.

SUMUP

Writes a summary of the number of aircraft damaged, the number of aircraft lost, NMCS aircraft hours and number NMCS aircraft to logical unit 6 at the end of each trial. Initializes variables SASLSTT, SASDAMT, SASNMHT, and SASNMAT to zero after the trial summary is written.

TIMES

Accumulates on-equipment task time (SASON), number of on-equipment tasks (CNTON), off-equipment task time (SASOFF), number of off-equipment tasks (CNTOFF), on-equipment personnel delay time (SASPER), number of on-equipment personnel delays (CNTPER), on-equipment AGE delay time (SASAGE), number of on-equipment AGE delays (CNTAGE), delay time for parts (SASPRT), number of delays for parts (CNTPRT), off-equipment personnel delay time (SASPEO), number of off-equipment delays for personnel (CNTPEO), off-equipment AGE delay time (SASEQP), and number of off-equipment AGE delays (CNTEQP).

PSTFLT

Accumulates the number of aircraft damaged (SASDAM).

CANNIB

Accumulates the number of cannibalizations (SASCAN).

KILLAC

Accumulates the number of aircraft lost (SASLST).

NORRPT

Accumulates the number of NMCS aircraft (SASNMA).

D. Variables Added to TSAR

Variable	Purpose
SAS1	Write "daily sorties flown and demanded" to logical unit 81. (CT 2/7)
SAS2	Write "aircraft lost" to logical unit 82. In addition, write summary of "aircraft lost" for each day and trial to logical unit 6. (CT 2/7)
SAS3	Write "aircraft damaged" to logical unit 83. In addition, write summary of "aircraft damaged" for each day and trial to logical unit 6. (CT 2/7)
SAS4	Write "cannibalizations" to logical unit 84. (CT 2/7)
SAS5	Write "NMCS hours" to logical unit 85. In addition, write summary of "NMCS hours" for each day and trial to logical unit 6. (CT 2/7)
SAS6	Write "NMCS aircraft" to logical unit 86. In addition, write summary of "NMCS aircraft" for each day and trial to logical unit 6. (CT 2/7)
SAS7	Write "on-equipment delays due to personnel" to logical unit 87. (CT 2/7)
SAS8	Write "off-equipment delays due to personnel" to logical unit 88. (CT 2/7)
SAS9	Write "on-equipment delays due to AGE" to logical unit 89. (CT 2/7)
SAS10	Write "off-equipment delays due to AGE" to logical unit 90. (CT 2/7)
SAS11	Write "delays due to spare parts" to logical unit 91. (CT 2/7)
SAS12	Write "on-equipment task time" to logical unit 92. (CT 2/7)
SAS13	Write "off-equipment repair time" to logical unit 93. (CT 2/7)

SASLST(ACTYPE,BASE)	Number of aircraft lost for each aircraft type by base. (subroutine KILLAC)
SASDAM(ACTYPE,BASE)	Number of aircraft damaged for each aircraft type by base. (subroutine PSTFLT)
SASPER(NOPEOP,BASE)	On-equipment personnel delay time (TTUS) for each personnel type by base. (Subroutine TIMES)
SASPEO(NOPEOP,BASE)	Off-equipment personnel delay time (TTUS) for each personnel type by base. (Subroutine TIMES)
SASAGE(NOAGE,BASE)	On-equipment AGE delay time (TTUS) for each AGE type by base. (Subroutine TIMES)
SASEQP(NOAGE,BASE)	Off-equipment AGE delay time (TTUS) for each AGE type by base. (Subroutine TIMES)
SASCAN(NOPART,ACTYPE,BASE)	Number of cannibalizations by part number for each type aircraft by base. (Subroutine CANNIB)
SASNMH(ACTYPE,BASE)	Number of NMCS aircraft hours for each base by aircraft type. (Subroutine NORRPT)
SASPRT(NOPART,BASE)	Delay time (TTUS) for each part type by base. (Subroutine TIMES)
SASDAMT(ACTYPE,BASE)	Summary of number of aircraft damaged by aircraft type for each base by trial. (Subroutine OUTPUT)
SASLSTT(ACTYPE,BASE)	Summary of number of aircraft lost by aircraft type for each base by trial. (Subroutine OUTPUT)
SASNMHT(ACTYPE,BASE)	Summary of number of NMCS aircraft hours by aircraft type for each base by trial. (Subroutine OUTPUT)
SASNMAT(ACTYPE,BASE)	Summary of number of NMCS aircraft by aircraft type for each base by trial. (subroutine OUTPUT)
SASON(BASE,SHOP)	On-equipment task time (TTUS) for each shop by base. (Subroutine TIMES)
SASOFF(BASE,SHOP)	Off-equipment task time (TTUS) for each shop by base. (subroutine TIMES)
CNTPER(NOPEOP,SHOP)	Number of on-equipment delays for each personnel type by shop. (Subroutine TIMES)

SNTPEO(NOPEOP,SHOP)	Number of off-equipment delays for each personnel type by shop. (Subroutine TIMES)
CNTAGE(NOAGE,BASE)	Number of on-equipment delays for each AGE type. (Subroutine TIMES)
CNTEQP(NOAGE,BASE)	Number of off-equipment delays for each AGE type by base. (Subroutine TIMES)
CNTPRT(NOPART,BASE)	Number of delays for each part type by base. (Subroutine TIMES)
CNTON(BASE,SHOP)	Number of on-equipment tasks for each shop by base. (Subroutine TIMES)
CNTOFF(BASE,SHOP)	Number of off-equipment tasks for each shop by base. (Subroutine TIMES)

E. ANALYSIS OF TSAR DATA

To ensure the accuracy of the output data as well as the compatibility of this data with the SAS software, SAS procedures were developed to test each of the output files. These procedures are representative of the kinds of analysis that may be done using SAS. They are not intended to be all inclusive but show that the output files are compatible with SAS. The outputs from the F-4G database (SAS inputs) are included as Appendix B. It should be noted that this TSAR run did not encounter any off-equipment delays for AGE. The data for on-equipment delays for AGE were used as the input for SAS.

1. INPUTS TO SAS

The output data from TSAR (input data to SAS) resides on disk and is read into SAS using the 'filename' and 'infile' statements. The actual SAS procedure files need not contain the data but only need to access it from the disk. This method makes changing data a matter of changing the data file name instead of deleting the old data and adding the new data into the SAS procedure file.

2. SAS PROCEDURES

There are assorted SAS procedures (PROCs) available to analyze the data from TSAR. It is unlikey that one would want to run all of these Procs at one time. The unneeded statements can be 'commented' out by placing an '*' in column one and entering a semi-colon following the comment or block of comments. The entire job can now be run with only the desired Procs being executed.

DESCRIPTIVE -- A complete set of SAS statements is available for modifying data. This permits great flexibility in formatting output.

ANALYTICAL -- The statistical analysis procedures in SAS range from simple statistics to complex multivarite techniques.

GRAPHIC -- SAS graphic procedures can be used to design displays showing information in meaningful pictures. Extensive title and footnote capabilities allow the user to explain the nature of the data or to annotate the display.

a. DESCRIPTIVE PROCS:

(1) PROC PRINT - prints a listing of the values of some or all of the variables in a SAS data set. Customized reports using options and statements are available. For example, with a BY statement PROC PRINT separates observations into groups defined by the BY variables. Numeric variable totals can be printed if a SUM statement is used.

(2) PROC TABULATE - displays descriptive statistics in hierachical tables. Each table cell belongs to a particular category of observations composed of crossing variable names. The statistic associated with each cell is calculated on values from all observations in that category.

PROC TABULATE provides:

* simple but powerful methods to create user-defined tables.

* a great degree of flexibility in classification of hierachies

* a variety of mechanisms for titling and formatting variables and procedure-generated statistics.

b. STATISTICAL PROC:

(1) PROC MEANS - can compute statistics for an entire SAS data set or for groups of observations in the data set. If the BY statement is used, MEANS calculates descriptive statistics separately for groups of observations. Each group is composed of observations having the same values of the variables used in the BY statement.

(2) PROC SUMMARY -similar to PROC MEANS except that MEANS produces subgroup statistics only when a BY statement is used. SUMMARY does not produce any printed output. The SUMMARY output data is typically printed with PROC PRINT or is input to a DATA step that extracts the desired information.

(3) PROC ANOVA - used to perform analysis of variance on balanced data. MEANS can be computed for any effect involving class variables. Several options are available for use with the MEANS procedure:

BON -- performs Bonferroni t tests of differences between means for all main-effect means in the MEANS statement.

DUNCAN -- performs Duncan's multiple range test on all main effect means given in the MEANS statement.

SCHEFFE -- perform[illegible] main-effect mean[illegible] the MEAN[illegible]

SNK -- performs the Student-Newman-Keuls multiple range test on all main-effect means in the MEANS statement.

TUKEY -- performs Tukey's studentized range test (HSD) on all main-effect means in the MEANS statement.

(4) PROCSTEPWISE - is most helpful for exploratory analysis because it can give the user insight into the relationship between the independent variables and the dependent variable. It provides five methods for stepwise regression. STEPWISE is useful when the best set of independent variables need to be found for inclusion in the regression model.

c. GRAPHIC PROCS:

(1) PROC PLOT -- graphs one variable against another, producing a printer plot. The coordinates of each point on the plot correspond to the value of two variables in one or more observations of the input data set.

(2) PROC CHART -- produces vertical and horizontal bar charts. These charts are useful for showing pictorially a variable's value or the relationship between two or more variables.

NOTE ---- Both of the above graphic procedures are available on SAS/GRAPH.* SAS/GRAPH is an enhanced graphics package that allows the user to introduce color and three dimensions into the output. Also, extensive title and footnote capabilities allow the user to explain the nature of the data and enhance the display.

3. SAS Manipulations

The following manipulations were performed on the output from the F-4G data base. The SAS Procs and a selected sample of results are contained in Appendix C.

TAPE 81 -- SORTIE FLOWN/DEMANDED

1. Average sorties flown by trial.
2. Average sorties flown by day.
3. Average sorties flown by aircraft.
4. Significant differences of sorties flown by trial.
5. Significant differences of sorties flown by day.

*SASGRAPH is a registered trademark of SAS Institute Inc., Carey, NC USA.

TAPE 82 -- AIRCRAFT LOST

1. Average aircraft lost by trial.
2. Average aircraft lost by day.
3. Significant differences of aircraft lost by trial.
4. Significant differences of aircraft lost by day.

TAPE 83 -- AIRCRAFT DAMAGED

1. Average aircraft damaged by trial.
2. Average aircraft damaged by day.
3. Significant differences of aircraft damaged by trial.
4. Significant differences of aircraft damaged by day.

TAPE 84 -- CANNIBALIZATIONS

1. Total number of parts canned by aircraft/trial.
2. Total number of parts canned by aircraft/day.
3. Total number of cannibalizations by aircraft/part.

TAPE 85 -- NMCS HOURS

1. Average number of NMCS hours by trial.
2. Average number of NMCS hours by day.
3. Significant number of NMCS hours by trial.
4. Significant number of NMCS hours by day.

TAPE 86 -- NMCS AIRCRAFT

1. Average number of NMCS aircraft by trial.
2. Average number of NMCS aircraft by day.
3. Significant number of NMCS aircraft by trial.
4. Significant number of NMCS aircraft by day.

TAPE 87 -- ON-EQUIP DELAY/PERSONNEL TYPE

1. Average number of delays by trial/personnel type.

2. Average number of delays by day/personnel type.

3. Significant number of delays by trial/personnel type.

4. Significant number of delays by day/personnel type.

TAPE 88 -- OFF-EQUIP DELAY/PERSONNEL TYPE

1. Average number of delays by trial/personnel type.

2. Significant number of delays by trial/personnel type.

TAPE 89 -- ON-EQUIP DELAY/AGE TYPE

1. Average number of delays by trial/AGE type.

2. Significant number of delays by trial/AGE type.

TAPE 90 -- OFF-EQUIP DELAY/AGE TYPE

1. Average number of delays by trial/AGE type.

2. Significant number of delays by trial/AGE type.

TAPE 91 -- DELAY TIME/PART TYPE

1. Average delay time by trial/part type.

2. Average delay time by day/part type.

3. Significant difference in mean delay time by trial/part type.

4. Significant differences in mean delay time by day/part type.

TAPE 92 -- ON-EQUIP DELAY TIME/SHOP NUMBER

1. Average delay time by trial/shop number.

2. Average delay time by day/shop number.

3. Significant difference in mean delay time by trial/shop number.

4. Significant difference in mean delay time by day/shop number.

TAPE 93 -- OFF-EQUIP DELAY TIME/SHOP NUMBER

1. Average delay time by trial/shop number.

2. Significant difference in mean delay time by trial/shop number.

4. EXAMPLES OF GRAPHIC PROCEDURES

The following graphic procedures were performed on the output from the F-4G data base. The SAS Procs and a selected sample of results are contained in Appendix D.

TAPE 81 -- SORTIES FLOWN/DEMANDED

1. Sorties flown by day/sorties demanded by day.

2. Sorties flown by day/trial.

TAPE 82 -- AIRCRAFT LOST

1. Aircraft lost by day/trial.

2. Aircraft lost by day.

TAPE 83 -- AIRCRAFT DAMAGED

1. Aircraft damaged by day/trial.

2. Aircraft damaged by trial.

TAPE 84 -- CANNIBALIZATIONS

1. Number of cannibalizations by day.

2. Number of cannibalizations by part/trial.

TAPE 85 -- NMCS HOURS

1. Number of NMCS hours by day.

2. Number of NMCS hours by day/trial.

TAPE 86 -- NMCS AIRCRAFT

1. Number of NMCS aircraft by day.

2. Number of NMCS aircraft by day/trial.

TAPE 87 - ON-EQUIP DELAY/PERSONNEL TYPE

1. Delay (minutes) by day/trial.

2. Delay (number) by personnel type day

TAPE 88 -- OFF-EQUIP DELAY/PERSONNEL TYPE

1. Delay (minutes) by day/trial.

2. Delay (number) by personnel type/day.

TAPE 89 -- ON-EQUIP DELAY/AGE TYPE

1. Delay (minutes) by day/trial.

2. Delay (number) by AGE type/day.

TAPE 90 -- OFF-EQUIP DELAY/AGE TYPE

1. Delay (minutes) by day/trial.

2. Delay (minutes) by AGE type/day.

TAPE 91 -- DELAY TIME/PART TYPE

1. Delay (minutes) by day/trial.

2. Delay (number) by part/day.

TAPE 92 -- ON-EQUIP DELAY TIME/SHOP

1. Delay (minutes) by day/trial.

2. Delay (number) by shop/day.

TAPE 93 -- OFF-EQUIP DELAY TIME/SHOP

1. Delay (minutes) by day/trial.

2. Delay (number) by shop/day.

APPENDIX A

TSAR IMPLEMENTATION CODE

TAB A

SUBROUTINE MANAGE

1. INSERT COMMON :

```
C
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
C
C*************************** COMMON SAS ******************************
C
      INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
     * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
     * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
C
      COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
     * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
     * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
     * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
     * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
     * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
C
C*************************** COMMON SAS ******************************
```

2. INSERT CODE AFTER LINES :

```
6116    REM = TD - 3*(TD/3)
        IF (REM .GT. 0) GO TO 6800
```

CODE TO BE INSERTED :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
           BOB = BASES(1,BASE)
           ATP = ACN(BOB,1)/512
           AIXD = ACN(BOB,2)
           BAS = AIXD-100*(AIXD/100)
           SASNMH(ATP,BAS) = SASNMH(ATP,BAS) + (3*NOR(BAS))
C       SAS OUTPUT PROCESSOR MOD
```

TAB B

SUBROUTINE INIT

1. INSERT COMMON :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
C
C***************************** COMMON SAS ******************************
C
       INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
      * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
      * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
      * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
C
       COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
      * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
      * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
      * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
      * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
      * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
      * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
C
C***************************** COMMON SAS ******************************
```

2. INSERT CODE AFTER LINES :

```
C    DIMENSION THE STORAGE ARRAYS FOR THE VARIOUS TASK REQUIREMENTS
C    NOTSK MAY NOT EXCEED 5000
         NOTSK  = 2500
         NOTSKA =  200
```

CODE TO BE INSERTED :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
          NOREP  = 1400
C       SAS OUTPUT PROCESSOR MOD
```

3. INSERT CODE AFTER LINES :

```
C    WXDAYS MAY NOT BE GREATER THAN 65
         WXDAYS =   65
C
```

CODE TO BE INSERTED :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
       JB = 1
3000   DO 3020 JA = 1,MAXT
        SASLST(JA,JB) = 0
        SASLSTT(JA,JB) = 0
        SASDAM(JA,JB) = 0
        SASDAMT(JA,JB) = 0
        SASNMA(JA,JB) = 0
        SASNMAT(JA,JB) = 0
        SASNMH(JA,JB) = 0
        SASNMHT(JA,JB) = 0
3020   CONTINUE
       IF (JB.EQ.MAXB) GO TO 3040
       JB = JB + 1
       GO TO 3000
3040   JB = 1
3060   DO 3080 JA = 1,NOPEOP
        SASPER(JA,JB) = 0
        CNTPER(JA,JB) = 0
        CNTPEO(JA,JB) = 0
        SASPEO(JA,JB) = 0
3080   CONTINUE
       IF (JB.EQ.MAXB) GO TO 3100
       JB = JB + 1
       GO TO 3060
3100   JB = 1
3120   DO 3140 JA = 1,NOAGE
        SASAGE(JA,JB) = 0
        CNTAGE(JA,JB) = 0
        CNTEQP(JA,JB) = 0
        SASEQP(JA,JB) = 0
```

```
3140  CONTINUE
      IF (JB.EQ.MAXB) GO TO 3160
      JB = JB + 1
      GO TO 3120
3160  JB = 1
3180  DO 3200 JA = 1,NOPART
       CNTPRT(JA,JB) = 0
       SASPRT(JA,JB) = 0
3200  CONTINUE
      IF (JB.EQ.MAXB) GO TO 3220
      JB = JB + 1
      GO TO 3180
3220  JB = 1
      JC = 1
3300  DO 3320 JA = 1,NOPART
      SASCAN(JA,JB,JC) = 0
3320  CONTINUE
      IF (JB.EQ.MAXT) GO TO 3340
      JB = JB + 1
      GO TO 3300
3340  IF (JC.EQ.MAXB) GO TO 3360
      JB = 1
      JC = JC + 1
      GO TO 3300
3360   JB = 1
3365   DO 3370 JA =1,30
        SASON(JB,JA) = 0
        SASOFF(JB,JA) = 0
        CNTON(JB,JA) = 0
        CNTOFF(JB,JA) = 0
3370   CONTINUE
        IF (JB.EQ.MAXB) GO TO 3375
        JB = JB + 1
        GO TO 3365
3375    CONTINUE
C      SAS OUTPUT PROCESSOR MOD
```

TAB B

SUBROUTINE INPUT

1. INSERT COMMON :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
C
C*************************** COMMON SAS ******************************
C
      INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
     * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
     * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
C
      COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
     * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
     * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
     * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
     * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
     * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
C
C*************************** COMMON SAS ******************************
```

REPLACE LINE WITH :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
  20      GO TO (21,22,29,292,294,295,270) J
C       SAS OUTPUT PROCESSOR MOD
```

2. INSERT CODE AFTER LINES :

```
 2931     FORMAT(' ',' ONLY 25 ITEMS MAY BE LISTED DURING',
     X                ' ANY SINGLE RUN WITH SUBRUTINE QUEUES' )
          GO TO 2
```

CODE TO BE INSERTED :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987

 270       SAS1 = D(1)
           SAS2 = D(2)
           SAS3 = D(3)
           SAS4 = D(4)
           SAS5 = D(5)
           SAS6 = D(6)
           SAS7 = D(7)
           SAS8 = D(8)
           SAS9 = D(9)
           SAS10 = D(10)
           SAS11 = D(11)
           SAS12 = D(12)
           SAS13 = D(13)
       IF (SAS1.NE.1) GO TO 880
       WRITE (81,1870)
       WRITE (81,1880)
 880   IF (SAS2.NE.1) GO TO 940
       WRITE (82,1890)
       WRITE (82,1900)
 940   IF (SAS3.NE.1) GO TO 990
       WRITE (83,1910)
       WRITE (83,1900)
 990   IF (SAS4.NE.1) GO TO 1083
       WRITE (84,1930)
       WRITE (84,1920)
 1083  IF (SAS5.NE.1) GO TO 1145
       WRITE (85,1940)
       WRITE (85,1950)
 1145  IF (SAS6.NE.1) GO TO 1205
       WRITE (86,1960)
       WRITE (86,1900)
 1205  IF (SAS7.NE.1) GO TO 1260
       WRITE (87,1970)
       WRITE (87,1990)
 1260  IF (SAS8.NE.1) GO TO 1320
       WRITE (88,1980)
       WRITE (88,1990)
 1320  IF (SAS9.NE.1) GO TO 1380
       WRITE (89,2000)
       WRITE (89,1990)
```

```
1380  IF (SAS10.NE.1) GO TO 1440
      WRITE (90,2010)
      WRITE (90,1990)
1440  IF (SAS11.NE.1) GO TO 1460
      WRITE (91,2020)
      WRITE (91,2030)
1460  IF (SAS12.NE.1) GO TO 1480
      WRITE (92,2040)
      WRITE (92,2050)
1480  IF (SAS13.NE.1) GO TO 1500
      WRITE (93,2060)
      WRITE (93,2050)
1500  CONTINUE
1870  FORMAT ('   SAS1=1  SORTIES FLOWN AND DEMANDED/BASE/ACFT TYPE/'
     * 'ACFT MISSION/PRIORITY')
1880  FORMAT ('  TRIAL  DAY  BASE  TYPE   MX    PRI FLOWN DEMANDED')
1890  FORMAT ('   SAS2=1  NUMBER OF ACFT LOST/BASE/ACFT TYPE')
1900  FORMAT ('   TRIAL   DAY    BASE  TYPE     #')
1910  FORMAT ('   SAS3=1 NUMBER OF ACFT DAMAGED/BASE/ACFT TYPE')
1920  FORMAT ('   TRIAL   DAY    BASE   TYPE PART      #')
1930  FORMAT ('   SAS4=1 NUMBER OF CANS/BASE/ACFT TYPE/PART')
1940  FORMAT ('   SAS5=1 NUMBER NMCS HOURS/BASE/ACFT TYPE')
1950  FORMAT ('   TRIAL   DAY    BASE  TYPE  HOURS')
1960  FORMAT ('   SAS6=1 NUMBER NMCS ACFT/BASE/ACFT TYPE')
1970  FORMAT ('   SAS7=1 ON-EQP DELAY TIME(MINUTES)/BASE/PERSONNEL',
     * ' TYPE')
1980  FORMAT ('   SAS8=1 OFF-EQP DELAY TIME (MINUTES)/BASE/PERSONNEL',
     * ' TYPE')
1990  FORMAT ('   TRIAL   DAY    BASE  TYPE  TIME      #')
2000  FORMAT ('   SAS9=1 ON-EQP DELAY TIME(MINUTES)/BASE/AGE TYPE')
2010  FORMAT ('   SAS10=1 OFF-EQP DELAY TIME(MINUTES)/BASE/AGE TYPE')
2020  FORMAT ('   SAS11=1 DELAY TIME(MINUTES)/BASE/PART NUMBER')
2030  FORMAT ('   TRIAL   DAY    BASE  PART  TIME      #')
2040  FORMAT ('   SAS12=1 ON-EQP TASK TIME(MINUTES)/BASE/SHOP')
2050  FORMAT ('   TRIAL   DAY    BASE  SHOP  TIME      #')
2060  FORMAT ('   SAS13=1 OFF-EQP REPAIR TIME(MINUTES)/BASE/SHOP')
C      SAS OUTPUT PROCESSOR MOD
```

TAB C

SUBROUTINE OUTPUT

1. INSERT COMMON :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
C
C***************************** COMMON SAS ******************************
C
      INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
     * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
     * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
C
      COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
     * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
     * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
     * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
     * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
     * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
C
C***************************** COMMON SAS ******************************
```

2 INSERT CODE AFTER LINES :

```
 6        CONTINUE
          MDAY = NODAY
          IF (IDAY .EQ. 2)  MDAY = (NODAY+1)/2
          IF (IDAY .EQ. 3)  MDAY = (NODAY+2)/3
C
C  MODIFY BREAK RATES AS APPROPRIATE
C
          IF  (VBREAK .EQ. 1)  CALL BREAK
C
          IF ((PRINT .LT. 1) .AND  (NTRIAL .EQ. 1))    GO TO 50
```

CODE TO BE INSERTED :

```
C   OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
      IF (SAS1 NE.1) GO TO 880
      JM = 1
      JA = 1
      JB = 1
800   DO 820 JP = 1,6
       IF (OUTPT1(1,JP,JM,JA,JB).NE.0)
     *  WRITE (81,1780) ITRIAL,NODAY,JB,JA,JM,JP,OUTPT1(2,JP,JM,JA,JB),
     *  OUTPT1(1,JP,JM,JA,JB)
820   CONTINUE
      IF (JM EQ MAXM) GO TO 840
      JM = JM + 1
      GO TO 800
840   IF (JA EQ MAXT) GO TO 860
      JM = 1
      JA = JA + 1
      GO TO 800
860   IF (JB EQ MAXB) GO TO 880
      JM = 1
      JA = 1
      JB = JB + 1
      GO TO 800
880   IF (SAS2 NE.1) GO TO 940
      JB = 1
900   DO 920 JA = 1,MAXT
        IF (SASLST(JA,JB).NE.0)
     *  WRITE (82,1800) ITRIAL,NODAY,JB,JA,SASLST(JA,JB)
        SASLSTT(JA,JB) = SASLSTT(JA,JB) + SASLST(JA,JB)
920   CONTINUE
      IF (JB.EQ.MAXB) GO TO 940
      JB = JB + 1
      GO TO 900
940   IF (SAS3 NE 1) GO TO 990
       JB = 1
960   DO 980 JA = 1,MAXT
        IF (SASDAM(JA,JB) NE.0)
     *  WRITE (83,1800) ITRIAL,NODAY,JB,JA,SASDAM(JA,JB)
        SASDAMT(JA,JB) = SASDAMT(JA,JB) + SASDAM(JA,JB)
980   CONTINUE
      IF (JB EQ MAXB) GO TO 990
      JB = JB +1
      GO TO 960
990   IF (SAS4 NE 1) GO TO 1080
```

```
      JA = 1
      JB = 1
1020  DO 1040 JP = 1,NOPART
      IF (SASCAN(JP,JA,JB) NE 0)
     *  WRITE (84,1805) ITRIAL,NODAY,JB,JA,JP,SASCAN(JP,JA,JB)
        SASCAN(JP,JA,JB) = 0
1040  CONTINUE
      IF (JA EQ MAXT) GO TO 1060
      JA = JA +1
      GO TO 1020
1060  IF (JB EQ MAXB) GO TO 1080
      JA = 1
      JB = JB + 1
      GO TO 1020
1080  IF (SAS5 NE 1) GO TO 1140
      JB = 1
1100  DO 1120 JA = 1,MAXT
        IF (SASNMH(JA,JB) NE 0)
     *  WRITE (85,1800) ITRIAL,NODAY,JB,JA,SASNMH(JA,JB)
        SASNMHT(JA,JB) = SASNMHT(JA,JB) + SASNMH(JA,JB)
1120  CONTINUE
      IF (JB EQ MAXB) GO TO 1140
      JB = JB +1
      GO TO 1100
1140  IF (SAS6 NE 1) GO TO 1200
      JB = 1
1160  DO 1180 JA = 1,MAXT
        IF (SASNMA(JA,JB) NE 0)
     *  WRITE (86,1800) ITRIAL,NODAY,JB,JA,SASNMA(JA,JB)
        SASNMAT(JA,JB) = SASNMAT(JA,JB) + SASNMA(JA,JB)
1180  CONTINUE
      IF (JB EQ MAXB) GO TO 1200
      JB = JB +1
      GO TO 1160
1200  IF (SAS7 NE 1) GO TO 1260
      JB = 1
1220  DO 1240 JA = 1,NOPEOP
        IF (SASPER(JA,JB) NE 0)
     *  WRITE (87,1810) ITRIAL,NODAY,JB,JA,(3*SASPER(JA,JB)),
     *  CNTPER(JA,JB)
        SASPER(JA,JB) = 0
1240  CONTINUE
      IF (JB EQ MAXB) GO TO 1260
      JB = JB +1
      GO TO 1220
1260  IF (SAS8 NE 1) GO TO 1320
      JB = 1
1280  DO 1300 JA = 1,NOPEOP
        IF (SASPEO(JA,JB) NE 0)
     *  WRITE (88,1810) ITRIAL,NODAY,JB,JA,(3*SASPEO(JA,JB)),
     *  CNTPEO(JA,JB)
        SASPEO(JA,JB) = 0
1300  CONTINUE
      IF (JB EQ MAXB) GO TO 1320
      JB = JB +1
      GO TO 1280
1320  IF (SAS9 NE 1) GO TO 1380
      JB = 1
1340  DO 1360 JA = 1,NOAGE
        IF (SASAGE(JA,JB) NE 0)
     *  WRITE (89,1815) ITRIAL,NODAY,JB,JA,(3*SASAGE(JA,JB)),
     *  CNTAGE(JA,JB)
        SASAGE(JA,JB) = 0
1360  CONTINUE
      IF (JB EQ MAXB) GO TO 1380
      JB = JB +1
      GO TO 1340
1380  IF (SAS10 NE 1) GO TO 1440
      JB = 1
1400  DO 1420 JA = 1,NOAGE
        IF (SASEQP(JA,JB) NE 0)
     *  WRITE (90,1815) ITRIAL,NODAY,JB,JA,(3*SASEQP(JA,JB)),
     *  CNTEQP(JA,JB)
        SASEQP(JA,JB) = 0
1420  CONTINUE
      IF (JB EQ MAXB) GO TO 1440
      JB = JB +1
1440  IF (SAS11 NE 1) GO TO 1500
      JB = 1
1460  DO 1480 JA = 1,NOPART
        IF (SASPRT(JA,JB) NE 0)
     *  WRITE (91,1820) ITRIAL,NODAY,JB,JA,(3*SASPRT(JA,JB)),
     *  CNTPRT(JA,JB)
        SASPRT(JA,JB) = 0
```

```
1480  CONTINUE
      IF (JB EQ MAXB) GO TO 1500
      JB = JB +1
      GO TO 1460
1500  IF (SAS12 NE 1) GO TO 1515
      JB = 1
1505  DO 1507 JA = 1,30
      IF (SASON(JB,JA) NE 0)
     *  WRITE (92 1815) ITRIAL,NODAY,JB,JA (3*SASON(JB,JA))
     *  CNTON(JB,JA)
      SASON(JB,JA) = 0
1507  CONTINUE
      IF (JB EQ MAXB) GO TO 1515
      JB = JB + 1
      GO TO 1505
1515  IF (SAS13 NE 1) GO TO 1530
      JB = 1
1520  DO 1525 JA = 1,30
      IF (SASOFF(JB,JA) NE 0)
     *  WRITE (93,1815) ITRIAL,NODAY,JB,JA,(3*SASOFF(JB,JA)),
     * CNTOFF(JB,JA)
      SASOFF(JB,JA) = 0
1525  CONTINUE
      IF (JB EQ MAXB) GO TO 1530
      JB = JB + 1
      GO TO 1520
1530  CONTINUE
1780  FORMAT (8(I5 2X  ))
1800  FORMAT (5(I6 2X  ))
1805  FORMAT (4(I6 2X  ),I6 4X  ,I6)
1810  FORMAT (3(I6 2X  ),I6 3X  ,2(I6  ))
1815  FORMAT (6(I6 2X  ))
1820  FORMAT (3(I6 2X  ),I6 4X  ,2(I6  ))
1840  FORMAT (  DAY ,I4,2X, TRIAL  ,I4,2X, BASE  ,I4,2X, AIRCRAFT
     * TYPE  ,I4,2X, # DAMAGED  ,I4,2X, # LOST  I4)
1850  FORMAT (  SAS END OF DAY SUMMARY )
1860  FORMAT (  DAY ,I4,2X, TRIAL  ,I4,2X, BASE  ,I4,2X, AIRCRAFT
     * TYPE  ,I4,2X, # NMCS  ,I4,2X, # NMCS HOURS  ,I4)

C       SAS OUTPUT PROCESSOR MOD


3  INSERT CODE AFTER LINE

    162     CONTINUE

   CODE TO BE INSERTED


C       DTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
        IF (SAS2 EQ 1 OR SAS3 EQ 1 OR SAS5 EQ 1 OR SAS6 EQ 1)
     *  WRITE (6,1850)
        IF (SAS2 NE 1 AND SAS3 NE 1) GO TO 170
        JB = 1
 165    DO 170 JA=1,MAXT
          IF (SASLST(JA,JB) NE 0 OR SASDAM(JA,JB) NE 0)
     *    WRITE (6,1840) (NODAY,ITRIAL,JB,JA,SASDAM(JA,JB),SASLST(JA,JB))
 170    CONTINUE
        IF (JB EQ MAXB) GO TO 175
        JB = JB + 1
        GO TO 165
 175    IF (SAS5 NE 1 AND SAS6 NE 1) GO TO 195
        JB = 1
 177    DO 180 JA=1,MAXT
        IF (SASNMA(JA,JB) NE 0 OR SASNMH(JA,JB) NE 0)
     *  WRITE (6,1860) (NODAY,ITRIAL,JB,JA,SASNMA(JA,JB),SASNMH(JA,JB))
 180    CONTINUE
        IF (JB EQ MAXB) GO TO 183
        JB = JB + 1
        GO TO 177
 183    JB = 1
 185    DO 190 JA=1,MAXT
            SASLST(JA,JB) = 0
            SASDAM(JA,JB) = 0
            SASNMH(JA,JB) = 0
            SASNMA(JA,JB) = 0
 190    CONTINUE
        IF (JB EQ MAXB) GO TO 195
        JB = JB + 1
        GO TO 185
 195    CONTINUE
C       SAS OUTPUT PROCESSOR MOD
```

```
TAB C

 SUBROUTINE SUMUP

  1  INSERT COMMON


     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
     C
     C************************** COMMON SAS ******************************
     C
           INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEOP,
          * SASCAN,SASNMH,SASNMA,SASPRT,SASDMT,SASLSTT,SASNMHT,SASNMAT,
          * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEOP,CNTPRT,CNTON,CNTOFF
     C
           COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEOP(99,63),
          * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
          * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
          * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
          * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
          * CNTEOP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
     C
     C************************** COMMON SAS ******************************


  2  INSERT CODE AFTER LINES

     132      CONTINUE
              WRITE(6,1068)  (NORMRS(B),B=1,NBASE)


     CODE TO BE INSERTED


     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
             IF (SAS2 EQ 1 OR SAS3 EQ 1 OR SAS5 EQ 1 OR SAS6 EQ 1)
          *   WRITE (6,2010) ITRIAL
             IF (SAS2 NE 1 AND SAS3 NE 1) GO TO 415
             JB = 1
      400    DO 410 JA=1,MAXT
             IF (SASLSTT(JA,JB) NE 0 OR SASDAMT(JA,JB) NE 0)
          *  WRITE (6,2000) (JB,JA,SASDAMT(JA,JB),SASLSTT(JA,JB))
             SASLSTT(JA,JB) = 0
             SASDAMT(JA,JB) = 0
      410    CONTINUE
             IF (JB EQ MAXB) GO TO 420
             JB = JB + 1
             GO TO 400
      415    IF (SAS5 NE 1 AND SAS6 NE 1) GO TO 135
      420    JB = 1
      440    DO 450 JA=1,MAXT
             IF (SASNMAT(JA,JB) NE 0 OR SASNMHT(JA,JB) NE 0)
          *  WRITE (6,2020) (JB,JA,SASNMAT(JA,JB),SASNMHT(JA,JB))
             SASNMAT(JA,JB) = 0
             SASNMHT(JA,JB) = 0
      450    CONTINUE
             IF (JB EQ MAXB) GO TO 135
             JB = JB + 1
             GO TO 440
     C       SAS OUTPUT PROCESSOR MOD


  3  INSERT CODE AFTER LINES

     1215     FORMAT( 0',20X,'FRACTION SORTIES OF AIRCRAFT TYPE ',I2,' THAT '
          X      'LAND WITH N UNSCHEDULED ON-EQUIPMENT BREAKS',/,' ',25X,9I7,
          X      '   >= ',I2,7X,' TOTAL     ','TASKS/SORTIE',/,' ',10X,
          X     'DEFERABLE ',5X,10F7.4,5X,F7.4,/,' ',10X,'IMMEDIATE ',5X,
          X     10F7.4,5X,F7.4,/,' ',12X,'EITHER',7X,10F7.4,5X,F7.4,4X,F7.3)


     CODE TO BE INSERTED


     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
     2000   FORMAT (' ',/,' BASE',I4,2X,'AIRCRAFT TYPE ',I4,2X,'# DAMAGED ',I4
          *,2X,'# LOST ',I4)
     2010     FORMAT (' SAS SUMMARY FOR TRIAL # ',I5)
     2020   FORMAT (' ',/,' BASE',I4,2X,'AIRCRAFT TYPE ',I4,2X,'# NMCS ',I4,2X
          *,'# NMCS HOURS ',I4)
     C       SAS OUTPUT PROCESSOR MOD
```

```
TAB C

 SUBROUTINE TIMES

  1  INSERT COMMON

     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH  1987
     C
     C**************************** COMMON SAS ****************************
     C
           INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
          * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
          * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
     C
           COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
          * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
          * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
          * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
          * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
          * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
     C
     C**************************** COMMON SAS ****************************

  2  INSERT CODE AFTER LINES

     C       KIND   =1 IS AN ON-EQUIPMENT TASK,
     C              =2 IS AN OFF-EQUIPMENT REPAIR,
     C              =3 IS AN AGE REPAIR JOB

     CODE TO BE INSERTED

     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH  1987
               NODAY = DAY(NOW)
     C       SAS OUTPUT PROCESSOR MOD

  3  INSERT CODE AFTER LINES

        10     OUTSHP(1,SHOP,BASE) = OUTSHP(1,SHOP,BASE) + 1
               OUTSHP(2,SHOP,BASE) = OUTSHP(2,SHOP,BASE) + TIME
               OUTSHP(3,SHOP,BASE) = OUTSHP(3,SHOP,BASE) + TIME*TIME

     CODE TO BE INSERTED

     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH  1987
               SASON(BASE,SHOP) = SASON(BASE,SHOP) + TIME
               CNTON(BASE,SHOP) = CNTON(BASE,SHOP) + 1
     C       SAS OUTPUT PROCESSOR MOD

  4  INSERT CODE AFTER LINES

        20     OUTSHP(4,SHOP,BASE) = OUTSHP(4,SHOP,BASE) + 1
               OUTSHP(5,SHOP,BASE) = OUTSHP(5,SHOP,BASE) + TIME
               OUTSHP(6,SHOP,BASE) = OUTSHP(6,SHOP,BASE) + TIME*TIME

     CODE TO BE INSERTED

     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH  1987
               SASOFF(BASE,SHOP) = SASOFF(BASE,SHOP) + TIME
               CNTOFF(BASE,SHOP) = CNTOFF(BASE,SHOP) + 1
     C       SAS OUTPUT PROCESSOR MOD
```

```
5  INSERT  ODE AFTER LINES

     100     IF (OUTPER(1,TYPE.BASE)  GT  2000000000)  RETURN
             OUTPER(1,TYPE.BASE) = OUTPER(1,TYPE.BASE) + ADD1
             OUTPER(2,TYPE.BASE) = OUTPER(2,TYPE.BASE) + ADD2

   CODE TO BE INSERTED

C        OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
           SASPER(TYPE.BASE) = SASPER(TYPE.BASE) + TIME
           CNTPER(TYPE.BASE) = CNTPER(TYPE.BASE) + 1
C        SAS OUTPUT PROCESSOR MOD

6  INSERT  ODE AFTER LINES

     200     OUTAGE(1,TYPE.BASE) = OUTAGE(1,TYPE.BASE) + ADD1
             OUTAGE(2,TYPE.BASE) = OUTAGE(2,TYPE.BASE) + ADD2

   CODE TO BE INSERTED

C        OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
           SASAGE(TYPE.BASE) = SASAGE(TYPE.BASE) + TIME
           CNTAGE(TYPE.BASE) = CNTAGE(TYPE.BASE) + 1
C        SAS OUTPUT PROCESSOR MOD

7  INSERT  ODE AFTER LINES

     300     OUTPRT(1,TYPE.BASE) = OUTPRT(1,TYPE.BASE) + ADD1
             OUTPRT(2,TYPE.BASE) = OUTPRT(2,TYPE.BASE) + ADD2

   CODE TO BE INSERTED

C        OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
           SASPRT(TYPE.BASE) = SASPRT(TYPE.BASE) + TIME
           CNTPRT(TYPE.BASE) = CNTPRT(TYPE.BASE) + 1
C        SAS OUTPUT PROCESSOR MOD

8  INSERT  ODE AFTER LINES

    1100     IF (OUTPEO(1,TYPE.BASE)  GT  2000000000)  RETURN
             OUTPEO(1,TYPE.BASE) = OUTPEO(1,TYPE.BASE) + ADD1
             OUTPEO(2,TYPE.BASE) = OUTPEO(2,TYPE.BASE) + ADD2

   CODE TO BE INSERTED

C        OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
           SASPEO(TYPE.BASE) = SASPEO(TYPE.BASE) + TIME
           CNTPEO(TYPE.BASE) = CNTPEO(TYPE.BASE) + 1
C        SAS OUTPUT PROCESSOR MOD

9  INSERT  ODE AFTER LINES

    1200     OUTEQP(1,TYPE.BASE) = OUTEQP(1,TYPE.BASE) + ADD1
             OUTEQP(2,TYPE.BASE) = OUTEQP(2,TYPE.BASE) + ADD2

   CODE TO BE INSERTED

C        OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
           SASEQP(TYPE.BASE) = SASEQP(TYPE.BASE) + TIME
           CNTEQP(TYPE.BASE) = CNTEQP(TYPE.BASE) + 1
C        SAS OUTPUT PROCESSOR MOD
```

```
TAB E

 SUBROUTINE PSTFLT

  1  INSERT COMMON


     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
     C
     C*************************** COMMON SAS ******************************
     C
           INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
          * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
          * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
     C
           COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
          * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
          * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
          * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
          * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
          * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
     C
     C*************************** COMMON SAS ******************************


  2  INSERT CODE AFTER LINES

      510     CONTINUE
              IF (BATTLE  NE  1)  GO TO 515
              ACN(NAC,35) = 1
              BASES(39,BASE) = BASES(39,BASE) + 1
              BASES(47,BASE) = BASES(47,BASE) + 1


     CODE TO BE INSERTED

     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
               SASDAM(ACTYPE,BASE) = SASDAM(ACTYPE,BASE) + 1
     C       SAS OUTPUT PROCESSOR MOD
```

```
TAB E

 SUBROUTINE CANNIB

  1  INSERT COMMON


     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
     C*************************** COMMON SAS ****************************
     C
           INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
          * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
          * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
     C
           COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
          * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
          * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
          * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
          * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
          * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
          * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
     C
     C*************************** COMMON SAS ****************************


  2  INSERT CODE AFTER LINES

     218       CFLAG = 1
               BASES(6,BASE) = BASES(6,BASE) + 1


     CODE TO BE INSERTED

     C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
               SASCAN(PART,ACTYPE,BASE) = SASCAN(PART,ACTYPE,BASE) + 1
     C       SAS OUTPUT PROCESSOR MOD
```

TAB H

SUBROUTINE KILLAC

1. INSERT COMMON

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
C
C***************************** COMMON SAS ******************************
C
      INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
     * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
     * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
C
      COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
     * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
     * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
     * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
     * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
     * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
C
C***************************** COMMON SAS ******************************
```

2 INSERT CODE AFTER LINES .

```
          IF (KILL .EQ. 1)  LOSTAC = LOSTAC + 1
C
          ACTYPE = ACN(NAC,1)/512
          AID = ACN(NAC,2)
          BASE = AID - 100*(AID/100)
          BASES(3,BASE) = BASES(3,BASE) - 1
          BASES(20+ACTYPE,BASE) = BASES(20+ACTYPE,BASE) - 8
          NF = BASES(1,BASE)
```

CODE TO BE INSERTED .

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
          SASLST(ACTYPE,BASE) = SASLST(ACTYPE,BASE) + 1
C       SAS OUTPUT PROCESSOR MOD
```

TAB J

SUBROUTINE NORRPT

1. INSERT COMMON :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
C**************************** COMMON SAS ******************************
C
      INTEGER *4 SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST,SASDAM,SASPER,SASPEO,SASAGE,SASEQP,
     * SASCAN,SASNMH,SASNMA,SASPRT,SASDAMT,SASLSTT,SASNMHT,SASNMAT,
     * SASON,SASOFF,CNTPER,CNTPEO,CNTAGE,CNTEQP,CNTPRT,CNTON,CNTOFF
C
      COMMON /SAS/ SAS1,SAS2,SAS3,SAS4,SAS5,SAS6,SAS7,SAS8,SAS9,SAS10,
     * SAS11,SAS12,SAS13,SASLST(9,63),SASDAM(9,63),SASEQP(99,63),
     * SASCAN(3199,9,63),SASNMH(9,63),SASNMA(9,63),SASPRT(3199,63),
     * SASPER(320,63),SASPEO(320,63),SASAGE(99,63),SASDAMT(9,63),
     * SASLSTT(9,63),SASNMHT(9,63),SASNMAT(9,63),SASON(9,30),
     * SASOFF(9,30),CNTPER(320,63),CNTPEO(320,63),CNTAGE(99,63),
     * CNTEQP(99,63),CNTPRT(3199,63),CNTON(9,30),CNTOFF(9,30)
C
C**************************** COMMON SAS ******************************
```

2 INSERT CODE AFTER LINES :

```
      ENTRY RPTNOR(PART,CRIT,NAC,BASE)
C     ************
C
        SIGNAL = BASE/100
        BASE = BASE - 100*SIGNAL
        IF (SIGNAL .EQ. 2)  GO TO 10
C
        IF ((NEO+1) .GE. LNOR) GO TO 90
C
C   WHEN A PRIOR HOLE REPORT IS BEING TRANSFERRED, RATHER THAN BEING
C   REPORTED FOR THE FIRST TIME, THIS FACT IS SIGNALED BY ADDING 100
C   TO THE NEW BASE NUMBER AND NO ACTION IS TAKEN TO ORDER A PART.
C
C   THIS SUBROUTINE IS ALSO CALLED BY "INITSK" WHENEVER A SERVICABLE
C   PART IS USED IN ORDER TO CALL "CONTRL" TO CHECK FOR A REPLACEMENT
C
        IF (CRIT  GT. 33)  CRIT = CRIT - 33
        IF (CRIT  GT. 33)  CRIT = CRIT - 33
C
C    RECORD THE NUMBER OF AIRCRAFT WITH HOLES, AND NOTE THE NUMBER
C    OF HOLES ON AIRCRAFT # NAC.
C
        ACN(NAC,22) = ACN(NAC,22) + 1
C
```

CODE TO BE INSERTED :

```
C       OTI SAS OUTPUT PROCESSOR MOD 1 MARCH, 1987
          ATP = ACN(NAC,1)/512
          AIXD = ACN(NAC,2)
          BAS = AIXD-100*(AIXD/100)
          SASNMA(ATP,BAS) = SASNMA(ATP,BAS) + 1
C       SAS OUTPUT PROCESSOR MOD
```

APPENDIX B

F-4G SAS INPUTS

SAS1=1 SORTIES FLOWN AND DEMANDED/BASE/ACFT TYPE/ACFT MISSION/PRIORITY

TRIAL	DAY	BASE	TYPE	MX	PRI	FLOWN	DEMANDED
01	01	01	01	01	01	04	06
01	01	01	01	01	03	00	08
01	01	01	01	02	01	02	08
01	01	01	01	02	02	00	10
01	01	01	01	03	01	04	06
01	01	01	01	03	02	00	10
01	01	01	01	03	03	16	76
01	01	01	01	04	01	02	04
01	01	01	01	04	02	01	10
01	01	01	01	04	03	26	36
01	02	01	01	01	01	06	06
01	02	01	01	01	02	06	20
01	02	01	01	01	03	00	08
01	02	01	01	02	01	06	08
01	02	01	01	03	01	06	06
01	02	01	01	03	03	21	76
01	02	01	01	04	01	04	04
01	02	01	01	04	03	19	36
01	03	01	01	01	01	06	06
01	03	01	01	01	02	10	10
01	03	01	01	01	03	00	08
01	03	01	01	02	01	03	08
01	03	01	01	02	02	00	10
01	03	01	01	03	01	06	06
01	03	01	01	03	03	11	76
01	03	01	01	04	01	04	04
01	03	01	01	04	03	13	36
01	04	01	01	01	01	06	06
01	04	01	01	01	03	14	40
01	04	01	01	02	01	06	08
01	04	01	01	02	03	00	16
01	04	01	01	03	01	06	06
01	04	01	01	03	03	03	64
01	04	01	01	04	01	02	04
01	04	01	01	04	03	12	40
01	05	01	01	01	01	04	06
01	05	01	01	01	03	11	40
01	05	01	01	02	01	06	08
01	05	01	01	02	03	07	16
01	05	01	01	03	01	06	06
01	05	01	01	03	03	04	64
01	05	01	01	04	01	04	04
01	05	01	01	04	03	08	40
01	06	01	01	01	01	06	06
01	06	01	01	01	03	08	40
01	06	01	01	02	01	06	08
01	06	01	01	02	03	02	16
01	06	01	01	03	01	06	06
01	06	01	01	03	03	02	64
01	06	01	01	04	01	02	04
01	06	01	01	04	03	06	40
01	07	01	01	01	01	04	06
01	07	01	01	01	03	09	40
01	07	01	01	02	01	04	08
01	07	01	01	02	03	07	16
01	07	01	01	03	01	06	06
01	07	01	01	03	03	04	64
01	07	01	01	04	01	02	04
01	07	01	01	04	03	08	40
01	08	01	01	01	01	04	06
01	08	01	01	01	03	08	40
01	08	01	01	02	01	04	08
01	08	01	01	02	03	08	16
01	08	01	01	03	01	06	06
01	08	01	01	03	03	06	64
01	08	01	01	04	01	04	04
01	08	01	01	04	03	07	40
01	09	01	01	01	01	04	06
01	09	01	01	01	03	08	40
01	09	01	01	02	01	04	08
01	09	01	01	02	03	03	16
01	09	01	01	03	01	06	06
01	09	01	01	03	03	02	64
01	09	01	01	04	01	00	04
01	09	01	01	04	03	08	40
01	10	01	01	01	01	04	06
01	10	01	01	01	03	04	40
01	10	01	01	02	01	02	08
01	10	01	01	02	03	06	16
01	10	01	01	03	01	06	06
01	10	01	01	03	03	03	64
01	10	01	01	04	01	02	04
01	10	01	01	04	03	04	40
02	01	01	01	01	01	06	06
02	01	01	01	01	03	02	08

02	01	01	01	02	01	00	08
02	01	01	01	02	02	00	10
02	01	01	01	03	01	03	06
02	01	01	01	03	02	00	10
02	01	01	01	03	03	14	76
02	01	01	01	04	01	02	04
02	01	01	01	04	02	01	10
02	01	01	01	04	03	26	36
02	02	01	01	01	01	06	06
02	02	01	01	01	02	06	20
02	02	01	01	01	03	04	08
02	02	01	01	02	01	04	08
02	02	01	01	03	01	06	06
02	02	01	01	03	03	13	76
02	02	01	01	04	01	02	04
02	02	01	01	04	03	13	36
02	03	01	01	01	01	06	06
02	03	01	01	01	02	02	10
02	03	01	01	01	03	00	08
02	03	01	01	02	01	04	08
02	03	01	01	02	02	00	10
02	03	01	01	03	01	06	06
02	03	01	01	03	03	14	76
02	03	01	01	04	01	02	04
02	03	01	01	04	03	06	36
02	04	01	01	01	01	04	06
02	04	01	01	01	03	10	40
02	04	01	01	02	01	02	08
02	04	01	01	02	03	00	16
02	04	01	01	03	01	06	06
02	04	01	01	03	03	02	64
02	04	01	01	04	01	00	04
02	04	01	01	04	03	11	40
02	05	01	01	01	01	04	06
02	05	01	01	01	03	07	40
02	05	01	01	02	01	06	08
02	05	01	01	02	03	04	16
02	05	01	01	03	01	06	06
02	05	01	01	03	03	04	64
02	05	01	01	04	01	02	04
02	05	01	01	04	03	06	40
02	06	01	01	01	01	04	06
02	06	01	01	01	03	08	40
02	06	01	01	02	01	06	08
02	06	01	01	02	03	00	16
02	06	01	01	03	01	06	06
02	06	01	01	03	03	02	64
02	06	01	01	04	01	02	04
02	06	01	01	04	03	04	40
02	07	01	01	01	01	04	06
02	07	01	01	01	03	08	40
02	07	01	01	02	01	06	08
02	07	01	01	02	03	04	16
02	07	01	01	03	01	06	06
02	07	01	01	03	03	03	64
02	07	01	01	04	01	02	04
02	07	01	01	04	03	04	40
02	08	01	01	01	01	06	06
02	08	01	01	01	03	08	40
02	08	01	01	02	01	02	08
02	08	01	01	02	03	08	16
02	08	01	01	03	01	06	06
02	08	01	01	03	03	06	64
02	08	01	01	04	01	02	04
02	08	01	01	04	03	06	40
02	09	01	01	01	01	04	06
02	09	01	01	01	03	06	40
02	09	01	01	02	01	02	08
02	09	01	01	02	03	07	16
02	09	01	01	03	01	05	06
02	09	01	01	03	03	07	64
02	09	01	01	04	01	02	04
02	09	01	01	04	03	04	40
02	10	01	01	01	01	04	06
02	10	01	01	01	03	07	40
02	10	01	01	02	01	02	08
02	10	01	01	02	03	07	16
02	10	01	01	03	01	06	06
02	10	01	01	03	03	07	64
02	10	01	01	04	01	02	04
02	10	01	01	04	03	07	40
03	01	01	01	01	01	06	06
03	01	01	01	01	03	00	08
03	01	01	01	02	01	00	08
03	01	01	01	02	02	00	10
03	01	01	01	03	01	04	06
03	01	01	01	03	02	00	10
03	01	01	01	03	03	10	76

03	01	01	01	04	01	02	04
03	01	01	01	04	02	01	10
03	01	01	01	04	03	27	36
03	02	01	01	01	01	06	06
03	02	01	01	01	02	08	20
03	02	01	01	01	03	00	08
03	02	01	01	02	01	04	08
03	02	01	01	03	01	06	06
03	02	01	01	03	03	08	76
03	02	01	01	04	01	01	04
03	02	01	01	04	03	21	36
03	03	01	01	01	01	06	06
03	03	01	01	01	02	05	10
03	03	01	01	01	03	00	08
03	03	01	01	02	01	08	08
03	03	01	01	02	02	00	10
03	03	01	01	03	01	06	06
03	03	01	01	03	03	08	76
03	03	01	01	04	01	04	04
03	03	01	01	04	03	15	36
03	04	01	01	01	01	05	06
03	04	01	01	01	03	09	40
03	04	01	01	02	01	08	08
03	04	01	01	02	03	02	16
03	04	01	01	03	01	06	06
03	04	01	01	03	03	02	64
03	04	01	01	04	01	04	04
03	04	01	01	04	03	07	40
03	05	01	01	01	01	04	06
03	05	01	01	01	03	10	40
03	05	01	01	02	01	08	08
03	05	01	01	02	03	00	16
03	05	01	01	03	01	06	06
03	05	01	01	03	03	00	64
03	05	01	01	04	01	02	04
03	05	01	01	04	03	04	40
03	06	01	01	01	01	04	06
03	06	01	01	01	03	10	40
03	06	01	01	02	01	04	08
03	06	01	01	02	03	05	16
03	06	01	01	03	01	06	06
03	06	01	01	03	03	04	64
03	06	01	01	04	01	04	04
03	06	01	01	04	03	08	40
03	07	01	01	01	01	04	06
03	07	01	01	01	03	11	40
03	07	01	01	02	01	06	08
03	07	01	01	02	03	05	16
03	07	01	01	03	01	06	06
03	07	01	01	03	03	05	64
03	07	01	01	04	01	04	04
03	07	01	01	04	03	08	40
03	08	01	01	01	01	04	06
03	08	01	01	01	03	07	40
03	08	01	01	02	01	06	08
03	08	01	01	02	03	07	16
03	08	01	01	03	01	06	06
03	08	01	01	03	03	03	64
03	08	01	01	04	01	02	04
03	08	01	01	04	03	08	40
03	09	01	01	01	01	06	06
03	09	01	01	01	03	08	40
03	09	01	01	02	01	06	08
03	09	01	01	02	03	08	16
03	09	01	01	03	01	06	06
03	09	01	01	03	03	04	64
03	09	01	01	04	01	04	04
03	09	01	01	04	03	07	40
03	10	01	01	01	01	04	06
03	10	01	01	01	03	10	40
03	10	01	01	02	01	04	08
03	10	01	01	02	03	04	16
03	10	01	01	03	01	06	06
03	10	01	01	03	03	02	64
03	10	01	01	04	01	02	04
03	10	01	01	04	03	09	40
04	01	01	01	01	01	04	06
04	01	01	01	01	03	00	08
04	01	01	01	02	01	00	08
04	01	01	01	02	02	00	10
04	01	01	01	03	01	04	06
04	01	01	01	03	02	00	10
04	01	01	01	03	03	12	76
04	01	01	01	04	01	02	04
04	01	01	01	04	02	01	10
04	01	01	01	04	03	32	36
04	02	01	01	01	01	06	06
04	02	01	01	01	02	06	20

04	02	01	01	01	03	00	08
04	02	01	01	02	01	05	08
04	02	01	01	03	01	06	06
04	02	01	01	03	03	07	76
04	02	01	01	04	01	04	04
04	02	01	01	04	03	24	36
04	03	01	01	01	01	04	06
04	03	01	01	01	02	01	10
04	03	01	01	01	03	00	08
04	03	01	01	02	01	06	08
04	03	01	01	02	02	02	10
04	03	01	01	03	01	06	06
04	03	01	01	03	03	12	76
04	03	01	01	04	01	04	04
04	03	01	01	04	03	22	36
04	04	01	01	01	01	04	06
04	04	01	01	01	03	12	40
04	04	01	01	02	01	08	08
04	04	01	01	02	03	03	16
04	04	01	01	03	01	06	06
04	04	01	01	03	03	04	64
04	04	01	01	04	01	04	04
04	04	01	01	04	03	08	40
04	05	01	01	01	01	04	06
04	05	01	01	01	03	14	40
04	05	01	01	02	01	06	08
04	05	01	01	02	03	00	16
04	05	01	01	03	01	06	06
04	05	01	01	03	03	02	64
04	05	01	01	04	01	04	04
04	05	01	01	04	03	09	40
04	06	01	01	01	01	06	06
04	06	01	01	01	03	12	40
04	06	01	01	02	01	05	08
04	06	01	01	02	03	03	16
04	06	01	01	03	01	06	06
04	06	01	01	03	03	03	64
04	06	01	01	04	01	02	04
04	06	01	01	04	03	04	40
04	07	01	01	01	01	06	06
04	07	01	01	01	03	12	40
04	07	01	01	02	01	06	08
04	07	01	01	02	03	05	16
04	07	01	01	03	01	06	06
04	07	01	01	03	03	03	64
04	07	01	01	04	01	02	04
04	07	01	01	04	03	08	40
04	08	01	01	01	01	04	06
04	08	01	01	01	03	11	40
04	08	01	01	02	01	06	08
04	08	01	01	02	03	06	16
04	08	01	01	03	01	06	06
04	08	01	01	03	03	07	64
04	08	01	01	04	01	04	04
04	08	01	01	04	03	09	40
04	09	01	01	01	01	06	06
04	09	01	01	01	03	08	40
04	09	01	01	02	01	06	08
04	09	01	01	02	03	07	16
04	09	01	01	03	01	06	06
04	09	01	01	03	03	01	64
04	09	01	01	04	01	02	04
04	09	01	01	04	03	07	40
04	10	01	01	01	01	04	06
04	10	01	01	01	03	07	40
04	10	01	01	02	01	07	08
04	10	01	01	02	03	07	16
04	10	01	01	03	01	06	06
04	10	01	01	03	03	03	64
04	10	01	01	04	01	02	04
04	10	01	01	04	03	06	40

SAS2=1 TRIAL	NUMBER DAY	OF ACFT LOST/BASE/ACFT TYPE BASE	TYPE	#
01	01	01	01	01
01	02	01	01	01
01	03	01	01	02
01	06	01	01	02
01	07	01	01	01
01	08	01	01	01
02	01	01	01	01
02	02	01	01	02
02	03	01	01	02
02	05	01	01	02
02	10	01	01	01
03	01	01	01	04
03	02	01	01	02
03	04	01	01	01
03	10	01	01	03
04	03	01	01	02
04	06	01	01	02
04	08	01	01	02
04	09	01	01	01

SAS3=1 NUMBER OF ACFT DAMAGED/BASE/ACFT TYPE

TRIAL	DAY	BASE	TYPE	#
01	01	01	01	01
01	05	01	01	01
01	06	01	01	01
02	02	01	01	01
03	01	01	01	02
03	03	01	01	01
03	04	01	01	01
04	01	01	01	03
04	02	01	01	02
04	03	01	01	01
04	05	01	01	02

SAS4=1 NUMBER OF CANS/BASE/ACFT TYPE/PART

TRIAL	DAY	BASE	TYPE	PART	#
02	07	01	01	0450	1

SAS5=1 NUMBER NMCS HOURS/BASE/ACFT TYPE

TRIAL	DAY	BASE	TYPE	HOURS
01	02	01	01	24
01	08	01	01	24
01	09	01	01	24
01	10	01	01	24
02	01	01	01	03
02	02	01	01	48
02	03	01	01	24
02	04	01	01	24
02	05	01	01	48
02	06	01	01	24
02	07	01	01	24
02	08	01	01	24
02	09	01	01	24
02	10	01	01	24
03	01	01	01	03
03	02	01	01	48
03	05	01	01	48
03	06	01	01	24
03	07	01	01	24
04	04	01	01	48
04	08	01	01	24
04	09	01	01	48

SAS6=1 TRIAL	NUMBER DAY	NMCS BASE	ACFT/BASE/ACFT TYPE TYPE	#
01	01	01	01	01
01	04	01	01	01
01	05	01	01	02
01	07	01	01	01
01	08	01	01	01
02	01	01	01	02
02	02	01	01	01
02	04	01	01	01
02	07	01	01	01
03	01	01	01	02
03	02	01	01	02
03	04	01	01	03
03	05	01	01	01
03	06	01	01	02
03	09	01	01	01
03	10	01	01	01
04	03	01	01	03
04	04	01	01	01
04	07	01	01	01
04	08	01	01	02
04	10	01	01	01

SAS7=1 ON-EQP DELAY TIME(MINUTES)/BASE/PERSONNEL TYPE

TRIAL	DAY	BASE	TYPE	TIME	#
01	01	01	001	117	16
01	01	01	012	120	1
01	01	01	015	675	17
01	01	01	030	96	3
01	01	01	031	144	16
01	01	01	042	96	3
01	01	01	043	1320	5
01	01	01	045	513	10
01	01	01	051	219	16
01	01	01	065	708	14
01	02	01	001	18	21
01	02	01	012	555	2
01	02	01	015	33	18
01	02	01	030	96	6
01	02	01	042	549	4
01	02	01	051	42	21
01	02	01	065	216	19
01	03	01	001	9	22
01	03	01	004	555	1
01	03	01	007	357	1
01	03	01	015	39	19
01	03	01	030	384	18
01	03	01	031	9	17
01	03	01	033	78	3
01	03	01	045	42	12
01	03	01	051	39	24
01	03	01	053	111	6
01	03	01	062	558	1
01	03	01	065	129	23
01	04	01	001	117	35
01	04	01	013	117	1
01	04	01	014	30	1
01	04	01	015	147	24
01	04	01	030	96	22
01	04	01	031	33	22
01	04	01	043	87	6
01	04	01	045	27	13
01	04	01	051	18	26
01	04	01	053	282	15
01	04	01	065	108	26
01	05	01	001	72	44
01	05	01	013	384	3
01	05	01	015	297	30
01	05	01	030	72	24
01	05	01	034	60	1
01	05	01	045	21	14
01	05	01	059	156	1
01	05	01	065	42	27
01	06	01	001	15	47
01	06	01	015	201	35
01	06	01	031	12	23
01	06	01	033	3	4
01	06	01	045	132	18
01	06	01	046	180	2
01	06	01	059	549	2
01	07	01	001	15	49
01	07	01	015	90	38
01	07	01	031	15	26
01	07	01	041	645	1
01	07	01	044	216	2
01	07	01	045	42	19
01	08	01	001	9	52
01	08	01	007	72	2
01	08	01	015	45	41
01	08	01	031	15	27
01	08	01	045	75	21
01	09	01	015	114	44
01	09	01	031	45	33
01	09	01	045	42	23
01	09	01	046	102	3
01	09	01	066	201	2
01	10	01	031	9	34
01	10	01	045	12	24
01	10	01	046	177	4
02	01	01	001	13209	62
02	01	01	015	576	57
02	01	01	030	96	27
02	01	01	031	13266	48
02	01	01	045	534	34
02	01	01	051	13248	37
02	01	01	065	633	38
02	01	01	066	114	3
02	02	01	001	18	65
02	02	01	007	117	3
02	02	01	015	111	59

02	02	01	030	72	29
02	02	01	056	30	1
02	02	01	065	54	40
02	03	01	001	12	68
02	03	01	003	3	1
02	03	01	012	30	3
02	03	01	013	30	4
02	03	01	014	42	2
02	03	01	030	480	44
02	03	01	051	12	38
02	03	01	059	189	3
02	04	01	001	51	75
02	04	01	013	306	7
02	04	01	028	15	1
02	04	01	030	216	51
02	04	01	051	9	39
02	04	01	053	30	17
02	04	01	059	132	4
02	04	01	063	18	1
02	04	01	064	120	2
02	04	01	065	174	46
02	05	01	011	561	1
02	05	01	051	6	40
02	05	01	053	48	18
02	05	01	058	1344	2
02	05	01	059	132	5
02	05	01	066	180	5
02	06	01	001	15	76
02	06	01	013	126	8
02	06	01	033	78	7
02	06	01	044	30	3
02	06	01	051	15	41
02	06	01	052	15	1
02	06	01	061	591	1
02	06	01	065	54	47
02	07	01	001	3	77
02	07	01	016	102	1
02	07	01	051	3	42
02	07	01	067	1116	2
02	08	01	015	42	60
02	08	01	043	108	8
02	08	01	051	15	46
02	08	01	065	84	49
02	09	01	051	6	47
02	09	01	065	42	50
02	10	01	001	15	78
02	10	01	051	9	49
02	10	01	053	162	22
02	10	01	064	459	8
02	10	01	065	27	51
03	01	01	001	51	89
03	01	01	015	558	74
03	01	01	017	9246	1
03	01	01	028	15	2
03	01	01	030	96	54
03	01	01	031	54300	67
03	01	01	045	495	43
03	01	01	051	15	52
03	01	01	065	618	61
03	02	01	030	24	55
03	02	01	045	54	44
03	02	01	051	12	53
03	02	01	059	60	6
03	02	01	065	27	62
03	03	01	007	357	4
03	03	01	008	240	2
03	03	01	030	144	59
03	03	01	045	105	47
03	03	01	053	78	25
03	03	01	059	216	7
03	03	01	065	27	63
03	03	01	066	60	6
03	04	01	003	117	2
03	04	01	030	576	77
03	04	01	037	72	1
03	04	01	042	681	5
03	04	01	044	48	5
03	04	01	045	240	55
03	04	01	051	12	56
03	04	01	057	1380	3
03	04	01	059	312	8
03	05	01	006	90	1
03	05	01	030	144	81
03	05	01	031	27	70
03	05	01	033	78	10
03	05	01	065	273	70
03	05	01	066	177	8
03	06	01	001	15	90

03	06	01	031	15	73
03	06	01	045	300	61
03	06	01	056	87	2
03	07	01	001	12	91
03	07	01	003	150	4
03	07	01	015	36	75
03	07	01	031	18	74
03	07	01	036	90	1
03	07	01	043	99	9
03	07	01	044	84	6
03	07	01	045	30	63
03	07	01	051	15	59
03	07	01	064	18	9
03	08	01	015	36	76
03	08	01	016	198	3
03	08	01	031	12	75
03	08	01	045	84	65
03	08	01	065	111	74
03	09	01	011	678	2
03	09	01	031	9	76
03	09	01	042	27	6
03	09	01	043	162	11
03	09	01	051	15	62
03	09	01	053	60	26
03	09	01	059	129	10
03	09	01	065	42	75
03	10	01	031	3	77
03	10	01	045	42	66
03	10	01	051	18	64
03	10	01	065	243	81
03	10	01	066	240	11
04	01	01	001	13689	105
04	01	01	015	498	85
04	01	01	017	14808	3
04	01	01	030	72	83
04	01	01	031	72	87
04	01	01	044	63	7
04	01	01	045	525	76
04	01	01	046	96	6
04	01	01	050	27	1
04	01	01	051	66	71
04	01	01	065	618	91
04	02	01	001	24	109
04	02	01	007	237	5
04	02	01	017	408	5
04	02	01	031	27	91
04	02	01	034	429	2
04	02	01	043	3	12
04	02	01	045	81	78
04	02	01	065	174	95
04	03	01	007	936	7
04	03	01	030	144	88
04	03	01	031	24	94
04	03	01	043	429	13
04	03	01	045	42	79
04	03	01	051	6	73
04	03	01	065	69	98
04	04	01	030	576	106
04	04	01	031	60	101
04	04	01	045	321	88
04	04	01	051	36	78
04	04	01	065	54	101
04	04	01	066	177	12
04	05	01	001	24	111
04	05	01	008	186	3
04	05	01	015	27	86
04	05	01	028	549	3
04	05	01	030	144	110
04	05	01	031	24	105
04	05	01	043	78	14
04	05	01	045	486	99
04	05	01	046	177	8
04	05	01	051	27	82
04	05	01	065	408	109
04	05	01	066	162	14
04	06	01	015	69	88
04	06	01	031	21	109
04	06	01	041	678	2
04	06	01	045	303	108
04	06	01	051	3	83
04	06	01	057	399	4
04	06	01	061	543	2
04	06	01	065	198	114
04	07	01	003	60	5
04	07	01	013	546	9
04	07	01	015	21	89
04	07	01	031	15	112
04	07	01	045	198	113

04	07	01	051	66	90
04	07	01	053	114	28
04	07	01	065	105	117
04	08	01	003	132	6
04	08	01	031	9	115
04	08	01	045	210	118
04	08	01	051	9	91
04	08	01	061	549	3
04	09	01	001	6	112
04	09	01	009	120	1
04	09	01	015	27	91
04	09	01	031	15	116
04	09	01	033	78	13
04	09	01	045	105	121
04	09	01	065	36	118
04	10	01	036	432	4
04	10	01	045	105	124
04	10	01	051	27	93

SAS8=1 OFF-EQP DELAY TIME (MINUTES)/BASE/PERSONNEL TYPE

TRIAL	DAY	BASE	TYPE	TIME	#
01	04	01	012	12	1
01	05	01	013	186	1
02	03	01	013	42	2
02	05	01	012	252	2
02	06	01	009	147	1
03	05	01	009	42	2
03	06	01	017	240	1
04	04	01	003	108	1
04	04	01	017	60	2
04	07	01	013	66	3

SAS9=1 ON-EQP DELAY TIME(MINUTES)/BASE/AGE TYPE

TRIAL	DAY	BASE	TYPE	TIME	#
01	01	01	06	09	01
01	02	01	06	09	02
01	08	01	06	09	03
02	01	01	06	27	06
03	07	01	06	09	07
04	01	01	06	27	10

SAS10=1 OFF-EQP DELAY TIME(MINUTES)/BASE/AGE TYPE

TRIAL	DAY	BASE	TYPE	TIME	#

SAS11=1 DELAY TIME(MINUTES)/BASE/PART NUMBER

TRIAL	DAY	BASE	PART	TIME	#
01	02	01	0302	609	1
01	04	01	0380	27	1
01	05	01	0371	552	1
01	05	01	0377	1065	1
01	08	01	0255	720	1
02	01	01	0211	3345	1
02	01	01	0606	2820	1
02	02	01	0249	639	1
02	02	01	0305	705	1
02	05	01	0299	1512	1
03	01	01	0128	12387	1
03	02	01	0057	621	1
03	02	01	0219	462	1
03	02	01	0250	807	1
03	02	01	0251	564	1
03	05	01	0201	576	1
03	05	01	0228	759	1
03	05	01	0230	753	1
03	05	01	0580	123	1
03	06	01	0232	621	1
03	06	01	0299	1215	2
03	07	01	0230	1521	2
03	09	01	0255	813	2
03	10	01	0301	627	1
04	03	01	0217	666	1
04	03	01	0616	144	1
04	04	01	0232	795	2
04	04	01	0254	915	1
04	04	01	0444	615	1
04	08	01	0268	642	1
04	09	01	0297	804	1
04	09	01	0388	756	1
04	10	01	0401	711	1

SAS12=1 ON-EQP TASK TIME(MINUTES)/BASE/SHOP

TRIAL	DAY	BASE	SHOP	TIME	#
01	01	01	19	300	02
01	02	01	10	585	01
01	02	01	19	300	03
01	03	01	01	234	01
01	03	01	19	426	05
01	03	01	28	723	01
01	04	01	01	3642	05
01	04	01	19	417	07
01	05	01	01	81	06
01	05	01	10	111	02
01	05	01	19	300	08
01	05	01	28	132	02
01	06	01	10	2070	03
01	06	01	19	303	09
01	07	01	10	2070	04
01	07	01	19	123	10
01	07	01	28	141	03
01	07	04	28	69	01
01	08	01	01	285	07
01	08	01	19	825	11
01	08	04	28	93	02
01	09	01	01	285	08
01	10	01	01	285	09
01	10	01	28	249	04
02	02	01	19	798	12
02	02	03	28	93	01
02	02	04	28	75	03
02	03	01	01	546	12
02	03	01	19	303	13
02	03	01	28	327	06
02	04	01	01	2883	13
02	04	01	10	2370	05
02	05	01	01	699	16
02	06	01	01	234	17
02	08	01	01	285	18
02	09	01	01	468	20
02	10	01	01	234	21
02	10	01	19	270	15
03	01	01	01	300	22
03	02	01	01	720	23
03	02	01	10	1410	07
03	02	01	19	1701	17
03	02	01	28	630	07
03	03	01	01	234	24
03	03	01	10	1 1	08
03	04	01	01	2883	25
03	04	01	10	585	09
03	04	01	19	297	18
03	04	04	28	1305	05
03	05	01	01	321	27
03	05	01	10	2070	10
03	05	01	19	1050	20
03	05	04	28	873	06
03	06	03	28	477	03
03	07	01	19	2250	24
03	07	03	28	171	04
03	08	01	01	285	28
03	09	01	01	639	31
03	09	01	19	1125	26
03	09	03	28	81	05
03	10	01	19	300	27
04	01	01	01	153	32
04	02	01	01	528	34
04	02	01	10	1782	13
04	02	01	19	300	28
04	02	01	28	597	08
04	03	01	01	2883	35
04	03	01	10	2070	14
04	03	01	28	231	09
04	03	04	28	1368	07
04	04	01	01	210	37
04	04	01	10	2895	16
04	04	01	19	2421	31
04	04	04	28	21	08
04	05	01	01	1224	40
04	05	01	10	585	17
04	06	01	01	165	41
04	06	01	10	585	18
04	06	01	19	165	32
04	06	04	28	453	09
04	07	01	19	183	33
04	08	01	10	201	19
04	08	01	19	1686	38
04	09	01	01	234	42
04	10	01	01	285	43

04	10	01	19	123	39
04	10	03	28	411	06

SAS13=1 OFF-EQP REPAIR TIME(MINUTES)/BASE/SHOP

TRIAL	DAY	BASE	SHOP	TIME	#
01	03	01	19	4956	02
01	05	01	19	3591	04
01	06	01	28	2226	01
01	07	01	19	4698	06
01	08	01	19	690	07
02	02	01	19	585	08
02	03	01	19	645	09
02	04	01	19	2370	10
02	05	01	19	696	11
03	02	01	10	621	01
03	02	01	19	1251	13
03	02	01	28	462	02
03	05	01	19	1941	14
03	05	04	28	1149	02
03	06	01	19	399	15
03	07	01	19	2622	16
03	08	01	19	1653	17
03	09	01	19	783	18
04	03	04	28	1329	04
04	04	01	19	3027	20
04	06	01	10	2643	02
04	06	01	19	2775	21
04	07	01	19	1713	22
04	08	01	19	582	23
04	09	01	19	4185	25
04	09	03	28	624	01

APPENDIX C

SAS MANIPULATIONS

```
FILENAME IN81 '[GORNTOC.SAS]F4GTAPE81.DAT';
DATA TAPE81;
  INFILE IN81;
  INPUT TNUM $ 4-5 DAY $ 10-11 BASENUM $ 17 ACTYPE $ 23 MISSTYP $ 29
        MISSPRI $ 35 SORTFLN 40-41 SORTDEM 46-47;
RUN;
*-------------------------------------------------------------------------*
|          PROCS TO PRINT AND PERFORM MEANS BY:                           |
|                                                                         |
|          1) TRIAL                                                       |
|          2) DAY                                                         |
|          3) TRIAL & DAY                                                 |
*-------------------------------------------------------------------------*  ;

PROC SORT; BY TNUM;
     PROC PRINT; BY TNUM;
     PROC MEANS; BY TNUM;
PROC SORT; BY DAY;
     PROC PRINT; BY DAY;
     PROC MEANS; BY DAY;
PROC SORT; BY TNUM DAY;
     PROC PRINT; BY TNUM DAY;
     PROC MEANS; BY TNUM DAY;

*-------------------------------------------------------------------------*
|     PROC TO PERFORM ANOVA TABLE AND FIND SIGNIFICANT DIFFERENCES BY      |
|             DAY AND TRIAL NUMBER WITH FULL INTERACTIONS                  |
*-------------------------------------------------------------------------*  ;

PROC ANOVA; CLASSES TNUM DAY;
     MODEL SORTFLN = TNUM DAY TNUM * DAY;
     MEANS TNUM | DAY/BON;

*---------------COMMENT OUT---------------------------*
*     MEANS TNUM | DAY/DUNCAN;                        *
*     MEANS TNUM | DAY/SCHEFFE;                       *
*     MEANS TNUM | DAY/SNK;                           *
*     MEANS TNUM | DAY/TUKEY;                         *
*-----------------------------------------------------*

*-------------------------------------------------------------------------*
|       PROCS TO PRINT TABULATIONS BY TRIAL, BASE, AND DAY                 |
*-------------------------------------------------------------------------*  ;

PROC TABULATE;
     CLASS TNUM BASENUM DAY;
     VAR SORTFLN SORTDEM;
     TABLE TNUM*DAY*(SUM MEAN STD), BASENUM*(SORTFLN SORTDEM);


*-------------------------------------------------------------------------*
|     PROC TO PERFORM PLOT PROCEDURE/ WITH SORTIES FLOWN BY DAY           |
|                                                                         |
|       The PLOT statement is in the form:                                |
|                 PLOT  vertical * horizontal = 'character'               |
|                                                                         |
|          /OVERLAY  combines plots on one page                           |
*-------------------------------------------------------------------------*  ;
PROC PLOT;
          PLOT SORTFLN * DAY = '*';
          PLOT SORTDEM * DAY = '@'/OVERLAY;


*-------------------------------------------------------------------------*
|               PROC TO PRODUCE BAR CHART OF SORTIES FLOWN BY             |
|                           BASE AND TRIAL                                |
```

```
*-------------------------------------------------------------------------* ;
PROC CHART;
     VBAR BASENUM / SUMVAR=SORTFLN GROUP=TNUM;
```

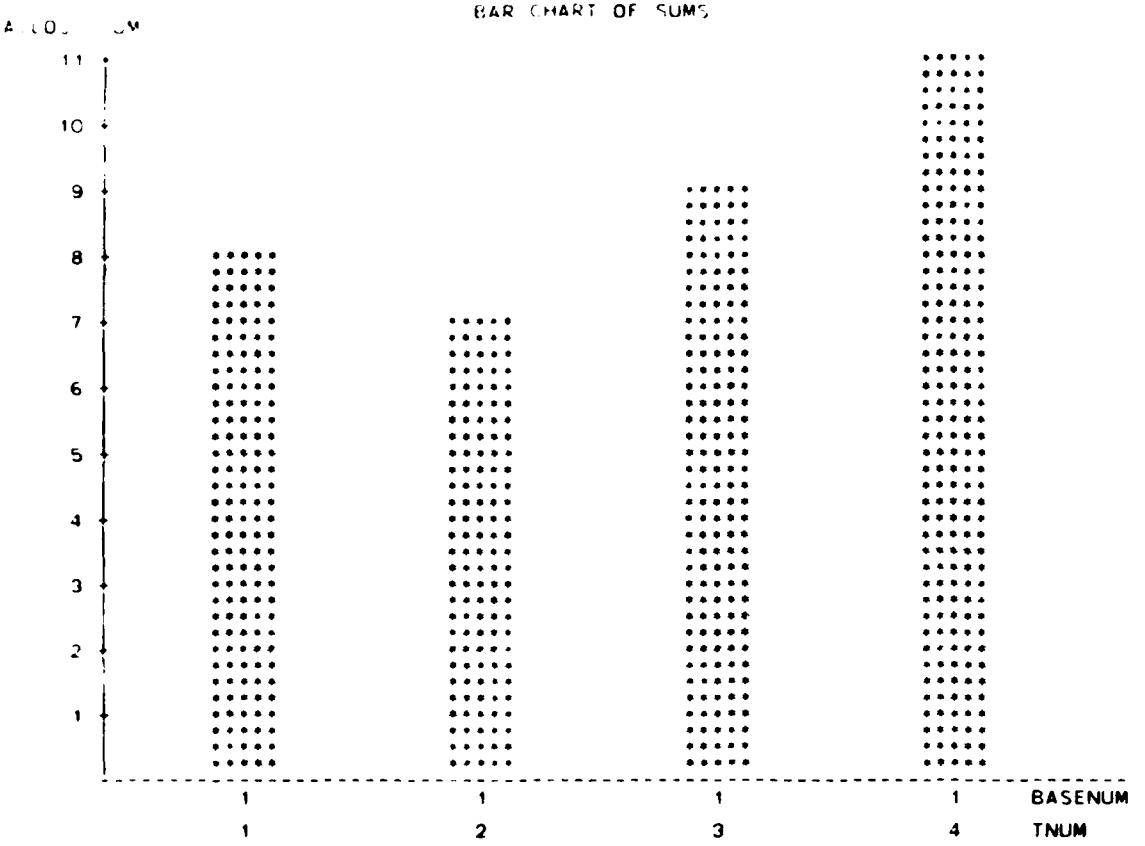
TNUM=01

OBS	DAY	BASENUM	ACTYPE	MISSTYP	MISSPRI	SORTFLN	SORTDEM
1	01	1	1	1	1	4	6
2	01	1	1	1	3	0	8
3	01	1	1	2	1	2	8
4	01	1	1	2	2	0	10
5	01	1	1	3	1	4	6
6	01	1	1	3	2	0	10
7	01	1	1	3	3	16	76
8	01	1	1	4	1	2	4
9	01	1	1	4	2	1	10
10	01	1	1	4	3	26	36
11	02	1	1	1	1	6	6
12	02	1	1	1	2	6	20
13	02	1	1	1	3	0	8
14	02	1	1	2	1	6	8
15	02	1	1	3	1	6	6
16	02	1	1	3	3	21	76
17	02	1	1	4	1	4	4
18	02	1	1	4	3	19	36
19	03	1	1	1	1	6	6
20	03	1	1	1	2	10	10
21	03	1	1	1	3	0	8
22	03	1	1	2	1	3	8
23	03	1	1	2	2	0	10
24	03	1	1	3	1	6	6
25	03	1	1	3	3	11	76
26	03	1	1	4	1	4	4
27	03	1	1	4	3	13	36
28	04	1	1	1	1	6	6
29	04	1	1	1	3	14	40
30	04	1	1	2	1	6	8
31	04	1	1	2	3	0	16
32	04	1	1	3	1	6	6
33	04	1	1	3	3	3	64
34	04	1	1	4	1	2	4
35	04	1	1	4	3	12	40
36	05	1	1	1	1	4	6
37	05	1	1	1	3	11	40
38	05	1	1	2	1	6	8
39	05	1	1	2	3	7	16
40	05	1	1	3	1	6	6
41	05	1	1	3	3	4	64
42	05	1	1	4	1	4	4
43	05	1	1	4	3	8	40
44	06	1	1	1	1	6	6
45	06	1	1	1	3	8	40
46	06	1	1	2	1	6	8
47	06	1	1	2	3	2	16
48	06	1	1	3	1	6	6
49	06	1	1	3	3	2	64
50	06	1	1	4	1	2	4
51	06	1	1	4	3	6	40
52	07	1	1	1	1	4	6
53	07	1	1	1	3	9	40

VARIABLE	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE	STD ERROR OF MEAN	SUM	VARIANCE	C V
TNUM=01									
SORTELN	83	5 66265[illegible]	4 58[illegible]154	0 00000000	26 00000000	0 50279183	470 0000000	20 98236850	80 892
SORTDEM	83	21 56626506	21 55208822	4 00000000	76 00000000	2 36564901	1790 0000000	464 49250661	99 934
TNUM=02									
SORTELN	83	4 98795181	3 93388648	0 00000000	26 00000000	0 43180014	414 0000000	15 47546283	78 868
SORTDEM	83	21 56626506	21 55208822	4 00000000	76 00000000	2 36564901	1790 0000000	464 49250661	99 934
TNUM=03									
SORTELN	83	5 54216867	4 21467010	0 00000000	27 00000000	0 46262014	460 0000000	17 76344402	76 047
SORTDEM	83	21 56626506	21 55208822	4 00000000	76 00000000	2 36564901	1790 0000000	464 49250661	99 934
TNUM=04									
SORTELN	83	5 91566265	5 07073098	0 00000000	32 00000000	0 55658503	491 0000000	25 71231267	85 717
SORTDEM	83	21 56626506	21 55208822	4 00000000	76 00000000	2 36564901	1790 0000000	464 49250661	99 934

```
FILENAME IN82 '[GORNTOC.SAS]F4GTAPE82.DAT';
DATA TAPE82;
  INFILE IN82;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 ACTYPE $ 27 ACLOSS 33-34;
RUN;
*-------------------------------------------------------*
|   THIS SECTION WILL COMPUTE AVERAGE ACFT LOST         |
|          BY TRIAL NUMBER AND BY DAY                   |
*-------------------------------------------------------* ;
PROC SORT; BY TNUM;
     PROC MEANS; BY TNUM;
PROC SORT; BY DAY;
     PROC MEANS; BY DAY;

*-------------------------------------------------------*
|    THIS SECTION WILL PRODUCE ANOVA TABLE AND FIND     |
|    SIGNIFICANT DIFFERENCES BY TRIAL NUMBER AND DAY    |
*-------------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY ;
     MODEL ACLOSS = TNUM DAY TNUM * DAY;
     MEANS TNUM | DAY /SNK;

*-------------------------------------------------------*
|    THIS SECTION WILL PRODUCE A BAR CHART OF ACFT LOST  |
|          BY BASENUM GROUPED BY TRIAL NUMBER           |
*-------------------------------------------------------* ;

PROC CHART;
     VBAR BASENUM/SUMVAR=ACLOSS GROUP=TNUM;

*-------------------------------------------------------*
|    THIS SECTION WILL PRODUCE TABLE OF ACFT LOST BY     |
|     TRIAL NUMBER AND DAY. THE TABLE WILL CONTAIN      |
|        SUM, MEAN, AND STD DEVIATION STATISTICS        |
*-------------------------------------------------------* ;

PROC TABULATE;
     CLASS TNUM DAY BASENUM;
     VAR ACLOSS;
     TABLE TNUM*(SUM MEAN STD), BASENUM*ACLOSS;
     TABLE DAY*(SUM MEAN STD), BASENUM*ACLOSS;
```

BAR CHART OF SUMS



```
FILENAME IN83 '[GORNTOC.SAS]F4GTAPE83.DAT';
DATA TAPE83;
  INFILE IN83;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 ACTYPE $ 27 ACDAM 33-34;
RUN;
*-------------------------------------------------------*
|   THIS SECTION WILL COMPUTE AVERAGE ACFT DAMAGED      |
|          BY TRIAL NUMBER AND BY DAY                   |
*-------------------------------------------------------* ;
PROC SORT; BY TNUM;
     PROC MEANS; BY TNUM;
PROC SORT; BY DAY;
     PROC MEANS; BY DAY;

*-------------------------------------------------------*
|    THIS SECTION WILL PRODUCE ANOVA TABLE AND FIND     |
|    SIGNIFICANT DIFFERENCES BY TRIAL NUMBER AND DAY    |
*-------------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY ;
     MODEL ACDAM  = TNUM DAY TNUM * DAY;
     MEANS TNUM | DAY /SNK;

*-------------------------------------------------------*
|  THIS SECTION WILL PRODUCE A BAR CHART OF ACFT DAMAGED |
|        BY BASENUM GROUPED BY TRIAL NUMBER             |
*-------------------------------------------------------* ;

PROC CHART;
     VBAR BASENUM/SUMVAR=ACDAM GROUP=TNUM;

*-------------------------------------------------------*
|  THIS SECTION WILL PRODUCE TABLE OF ACFT DAMAGED BY   |
|     TRIAL NUMBER AND DAY. THE TABLE WILL CONTAIN      |
|        SUM, MEAN, AND STD DEVIATION STATISTICS        |
*-------------------------------------------------------* ;

PROC TABULATE;
     CLASS TNUM DAY BASENUM;
     VAR ACDAM;
     TABLE TNUM*(SUM MEAN STD), BASENUM*ACDAM;
     TABLE DAY*(SUM MEAN STD), BASENUM*ACDAM;
```

```
FILENAME IN84 '[GORNTOC.SAS]F4Gtape84.DAT';
DATA TAPE84;
  INFILE IN84;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 ACTYPE $ 27 PART $ 32-34
        CANNAMT 40-41;
RUN;
*-------------------------------------------------------*
|    PROC TO AVERAGE NUMBER OF CANN'S BY TRIAL NUMBER   |
|                  AND BY PART                          |
*-------------------------------------------------------* ;

PROC SORT; BY TNUM PART;
           PROC MEANS; BY TNUM PART;

*-------------------------------------------------------*
|    PROC TO TOTAL NUMBER OF CANN'S BY TRIAL NUMBER     |
|                  AND BY DAY                           |
*-------------------------------------------------------* ;

PROC TABULATE;
     TITLE 'NUMBER OF CANNS BY TRIAL NUM / DAY';
     CLASS DAY TNUM;
     VAR CANNAMT;
     LABEL TNUM = 'TRIAL NUM';
     KEYLABEL  SUM = 'TOTAL'
               STD = 'STD DEV';
     TABLE TNUM * DAY , CANNAMT * (SUM MEAN);

*-------------------------------------------------------*
|    PROC TO TOTAL NUMBER OF CANN'S BY TRIAL NUMBER     |
|                  AND BY PART                          |
*-------------------------------------------------------* ;

PROC TABULATE;
     TITLE 'NUMBER OF CANNS BY TRIAL NUM / PART NUM';
     CLASS TNUM PART;
     VAR CANNAMT;
     LABEL TNUM = 'TRIAL NUM';
     KEYLABEL  SUM = 'TOTAL'
               STD = 'STD DEV';
     TABLE TNUM * PART , CANNAMT * (SUM MEAN);
```

```
DATA TAPE85;
  INFILE IN85;
  INPUT TNUM $ 5-6 ACTYPE $ 12-13 BASENUM $ 20 DAY $ 27 NMCSHR 32-34;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NMCS HOURS BY TRIAL   |
|                     AND DAY                     |
*-------------------------------------------------* ;

PROC SORT; BY TNUM;
           PROC MEANS; BY TNUM;
PROC SORT; BY DAY;
           PROC MEANS; BY DAY;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|     TRIAL AND DAY WITH FULL INTERACTIONS        |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY ;
     MODEL NMCSHR = TNUM  DAY TNUM*DAY;
     MEANS TNUM  DAY TNUM|DAY/SNK;
```

```
FILENAME IN86 '[GORNTOC.SAS]F4Gtape86.DAT';
DATA TAPE86;
  INFILE IN86;
  INPUT TNUM $ 5-6 ACTYPE $ 12-13 BASENUM $ 20 DAY $ 27
        NMCSAC 33-34;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NMCS ACFT BY TRIAL,   |
|            DAY, AND BASE NUMBER                 |
*-------------------------------------------------* ;

PROC SORT; BY TNUM;
           PROC MEANS; BY TNUM;
PROC SORT; BY DAY;
           PROC MEANS; BY DAY;
PROC SORT; BY BASENUM;
           PROC MEANS; BY BASENUM;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|     TRAIL, DAY, AND BASE NUMBER WITH FULL       |
|                 INTERACTIONS                    |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY BASENUM;
     MODEL NMCSAC = TNUM  DAY BASENUM TNUM*DAY*BASENUM;
     MEANS TNUM  DAY BASENUM TNUM|DAY|BASENUM/SNK;
```

```
FILENAME IN87 '[GORNTOC.SAS]F4Gtape87.DAT';
DATA TAPE87;
  INFILE IN87;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 PERSON $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NUMBER OF DELAYS BY   |
|           1) TNUM/PERSONNEL TYPE                |
|           2) DAY/PERSONNEL TYPE                 |
*-------------------------------------------------* ;

PROC SORT; BY TNUM PERSON;
           PROC MEANS; BY TNUM PERSON;
PROC SORT; BY DAY PERSON;
           PROC MEANS; BY DAY PERSON;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|           1) TRIAL                              |
|           2) DAY                                |
|           3) PERSONNEL TYPE                     |
|           4) COMBINATIONS OF THE ABOVE          |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY PERSON;
     MODEL DELNUM = TNUM  DAY PERSON TNUM*DAY*PERSON;
     MEANS TNUM  DAY PERSON TNUM|DAY|PERSON/SNK;
```

```
FILENAME IN88 '[GORNTOC.SAS]F4GTAPE88.DAT';
DATA TAPE88;
  INFILE IN88;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 PERSON $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|      PROCS TO FIND AVERAGE NUMBER OF DELAYS BY  |
|            1) TNUM/PERSONNEL TYPE               |
|            2) DAY/PERSONNEL TYPE                |
*-------------------------------------------------* ;

PROC SORT; BY TNUM PERSON;
           PROC MEANS; BY TNUM PERSON;
PROC SORT; BY DAY PERSON;
           PROC MEANS; BY DAY PERSON;

*-------------------------------------------------*
|      PROCS TO FIND SIGNIFICANT DIFFERENCES BY   |
|            1) TRIAL                             |
|            2) DAY                               |
|            3) PERSONEL TYPE                     |
|            4) COMBINATIONS OF THE ABOVE         |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY PERSON;
     MODEL DELNUM = TNUM  DAY PERSON TNUM*DAY*PERSON;
     MEANS TNUM  DAY PERSON TNUM|DAY|PERSON/SNK;
```

```
FILENAME IN89 '[GORNTOC.SAS]F4GTAPE89.DAT';
DATA TAPE89;
  INFILE IN89;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 AGE $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NUMBER OF DELAYS BY   |
|           1) TNUM/AGE TYPE                      |
|           2) DAY/AGE TYPE                       |
*-------------------------------------------------* ;

PROC SORT; BY TNUM AGE;
           PROC MEANS; BY TNUM AGE;
PROC SORT; BY DAY AGE;
           PROC MEANS; BY DAY AGE;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|           1) TRIAL                              |
|           2) DAY                                |
|           3) AGE TYPE                           |
|           4) COMBINATIONS OF THE ABOVE          |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY AGE;
     MODEL DELNUM = TNUM  DAY AGE TNUM*DAY*AGE;
     MEANS TNUM  DAY AGE TNUM|DAY|AGE/SNK;
```

```
FILENAME IN90 '[GORNTOC.SAS]F4GTAPE90.DAT';
DATA TAPE90;
  INFILE IN90;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 AGE $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NUMBER OF DELAYS BY   |
|           1) TNUM/AGE TYPE                      |
|           2) DAY/AGE TYPE                       |
*-------------------------------------------------* ;

PROC SORT; BY TNUM AGE;
           PROC MEANS; BY TNUM AGE;
PROC SORT; BY DAY AGE;
           PROC MEANS; BY DAY AGE;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|           1) TRIAL                              |
|           2) DAY                                |
|           3) AGE TYPE                           |
|           4) COMBINATIONS OF THE ABOVE          |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY AGE;
     MODEL DELNUM = TNUM  DAY AGE TNUM*DAY*AGE;
     MEANS TNUM  DAY AGE TNUM|DAY|AGE/SNK;
```

```
FILENAME IN91 '[GORNTOC.SAS]F4GTAPE91.DAT';
DATA TAPE91;
  INFILE IN91;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 PART $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NUMBER OF DELAYS BY   |
|           1) TNUM/PART                          |
|           2) DAY/PART                           |
*-------------------------------------------------* ;

PROC SORT; BY TNUM PART;
           PROC MEANS; BY TNUM PART;
PROC SORT; BY DAY PART;
           PROC MEANS; BY DAY PART;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|           1) TRIAL                              |
|           2) DAY                                |
|           3) PART NUMBER                        |
|           4) COMBINATIONS OF THE ABOVE          |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY PART;
     MODEL DELNUM = TNUM  DAY PART TNUM*DAY*PART;
     MEANS TNUM  DAY PART TNUM|DAY|PART/SNK;
```

```
FILENAME IN92 '[GORNTOC.SAS]F4GTAPE92.DAT';
DATA TAPE92;
  INFILE IN92;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 SHOP $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NUMBER OF DELAYS BY   |
|           1) TNUM/SHOP NUMBER                   |
|           2) DAY/SHOP NUMBER                    |
*-------------------------------------------------* ;

PROC SORT; BY TNUM SHOP;
           PROC MEANS; BY TNUM SHOP;
PROC SORT; BY DAY SHOP;
           PROC MEANS; BY DAY SHOP;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY    |
|           1) TRIAL                              |
|           2) DAY                                |
|           3) SHOP NUMBER                        |
|           4) COMBINATIONS OF THE ABOVE          |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY SHOP;
     MODEL DELNUM = TNUM  DAY SHOP TNUM*DAY*SHOP;
     MEANS TNUM  DAY SHOP TNUM|DAY|SHOP/SNK;
```

```
FILENAME IN93 '[GORNTOC.SAS]F4GTAPE93.DAT';
DATA TAPE93;
  INFILE IN93;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 SHOP $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------*
|     PROCS TO FIND AVERAGE NUMBER OF DELAYS BY   |
|           1) TNUM/SHOP NUMBER                   |
*-------------------------------------------------* ;

PROC SORT; BY TNUM SHOP;
           PROC MEANS; BY TNUM SHOP;

*-------------------------------------------------*
|     PROCS TO FIND SIGNIFICANT DIFFERENCES BY     |
|           1) TRIAL                              |
|           2) DAY                                |
|           3) SHOP NUMBER                        |
|           4) COMBINATIONS OF THE ABOVE          |
*-------------------------------------------------* ;

PROC ANOVA; CLASSES TNUM DAY SHOP;
     MODEL DELNUM = TNUM  DAY SHOP TNUM*DAY*SHOP;
     MEANS TNUM  DAY SHOP TNUM|DAY|SHOP/SNK;
```

APPENDIX D

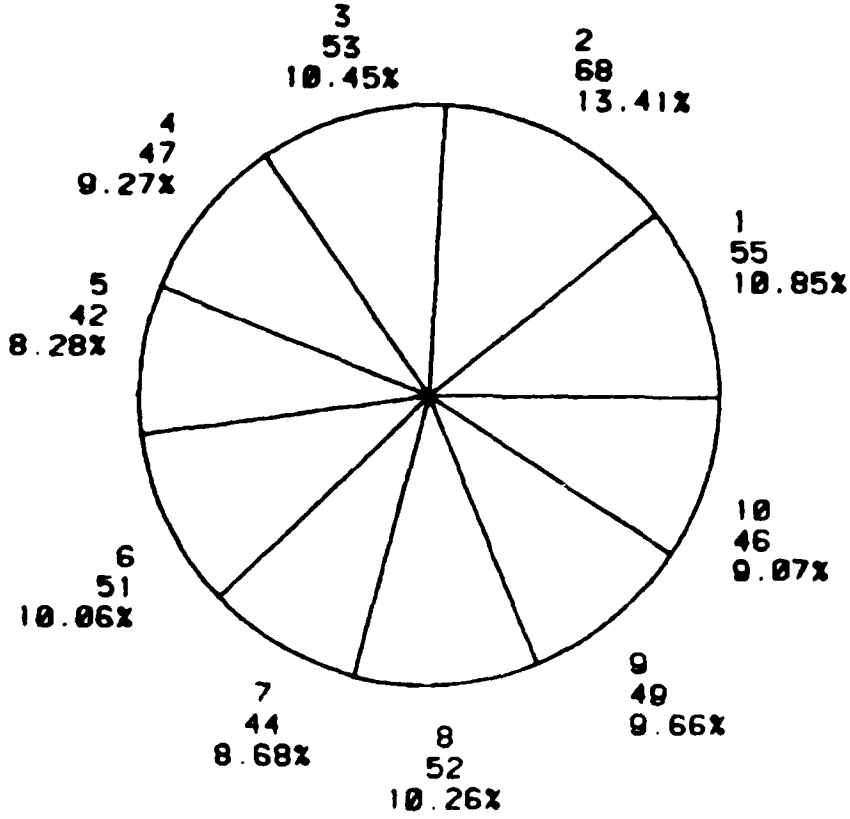
SAS GRAPHIC PROCEDURES

```
FILENAME IN81 '[GORNTOC.SAS]F4GTAPE81.DAT';
DATA TAPE81;
  INFILE IN81;
  INPUT (TNUM DAY BASENUM ACTYPE MISSTYP MISSPRI SORTFLN SORTDEM )
        (5.,6.,6.,6.,6.,6.,6.,6.);
RUN;

*-------------------------------------------------------*
|     THIS SECTION WILL PRODUCE PIE CHARTS,             |
|       THE SLICES WILL CORRESPOND TO DAY,              |
|       WITH EACH NEW PIE CORRESPONDING TO              |
|                 THE TRIAL NUMBER                      |
*-------------------------------------------------------* ;
PROC SORT; BY TNUM DAY;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'TSAR SORTIES FLOWN BY DAY/TRIAL';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE DAY/SUMVAR   = SORTFLN
                DISCRETE
                PCT      = OUTSIDE;
RUN;


*-------------------------------------------------------*
       THIS SECTION WILL PRODUCE X-Y PLOTS
        OF SORTIES FLOWN BY DAY OVERLAYED
          WITH SORTIES DEMANDED BY DAY.
   A NEW PLOT WILL BE DRAWN FOR EACH TRIAL NUMBER
*-------------------------------------------------------* ;

PROC SORT; BY TNUM MISSTYP MISSPRI DAY ;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'TSAR SORTIES FLOWN/DEMANDED';
TITLE2 F=TRIPLEX 'BY TRIAL/MISSION TYPE/MISSION PRIORITY';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
SYMBOL2   W=1 I=JOIN V=+;
PROC GPLOT;
BY TNUM MISSTYP MISSPRI;
       PLOT SORTFLN   DAY =1
            SORTDEM * DAY =2  /OVERLAY;
RUN;
```

TSAR SORTIES FLOWN BY DAY/TRIAL

TNUM=1

SUM OF SORTFLN BY DAY



AIRCRAFT <--> F-4C

```
FILENAME IN82 '[GORNTOC.SAS]F4GTAPE82.DAT';
DATA TAPE82;
  INFILE IN82;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 ACTYPE $ 27 ACLOSS 33-34;
RUN;

*-------------------------------------------------------*
|     THIS SECTION WILL PRODUCE PIE CHARTS,             |
|       THE SLICES WILL CORRESPOND TO DAY,              |
|       WITH EACH NEW PIE CORRESPONDING TO              |
|                 THE TRIAL NUMBER                      |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ACFT LOST BY DAY/TRIAL';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE DAY/SUMVAR  = ACLOSS
                PCT     = OUTSIDE;
RUN;

*-------------------------------------------------------*
|       THIS SECTION WILL PRODUCE X-Y PLOTS             |
|            OF AIRCRAFT LOST BY DAY                    |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL NUMBER       |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ACFT LOST BY DAY';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM;
       PLOT ACLOSS * DAY =1;
RUN;
```

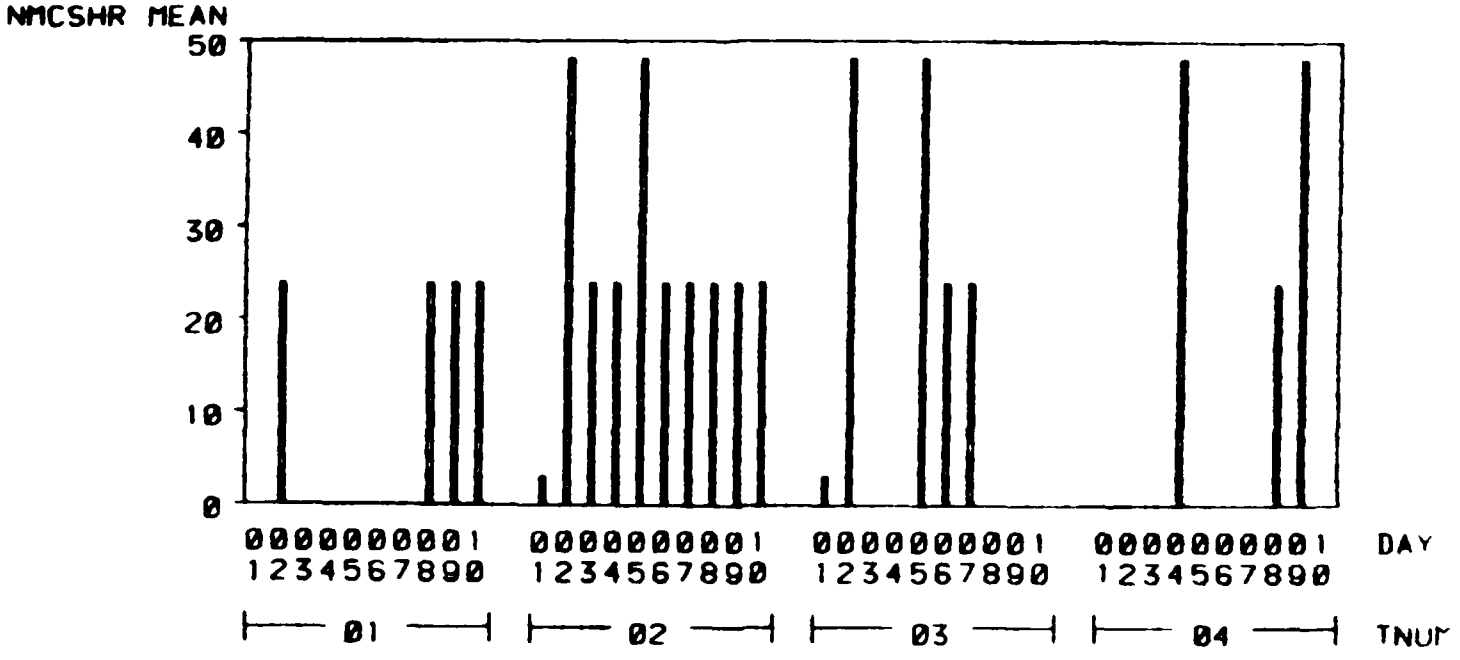
```
FILENAME IN83 '[GORNTOC.SAS]F4GTAPE83.DAT';
DATA TAPE83;
  INFILE IN83;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 ACTYPE $ 27 ACDAM 33-34;
RUN;

*-------------------------------------------------------*
|     THIS SECTION WILL PRODUCE PIE CHARTS,             |
|       THE SLICES WILL CORRESPOND TO DAY,              |
|       WITH EACH NEW PIE CORRESPONDING TO              |
|                 THE TRIAL NUMBER                      |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ACFT DAMAGED BY DAY/TRIAL';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE DAY/SUMVAR  = ACDAM
                 PCT    = OUTSIDE;
RUN;

*-------------------------------------------------------*
|      THIS SECTION WILL PRODUCE X-Y PLOTS              |
|            OF AIRCRAFT DAMAGED BY DAY                |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL NUMBER       |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ACFT DAMAGED BY DAY';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM;
       PLOT ACDAM * DAY =1;
RUN;
```

```
FILENAME IN84 '[GORNTOC.SAS]F4Gtape84.DAT';
DATA TAPE84;
  INFILE IN84;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 ACTYPE $ 27 PART $ 32-34
        CANNAMT 40-41;
RUN;

*-------------------------------------------------------*
|         THIS SECTION WILL PRODUCE BAR CHARTS,         |
|        EACH BAR WILL REPRESENT TOTAL NUMBER OF        |
|                   CANN'S BY DAY                       |
|              NEW GRAPH PRODUCED FOR EACH TRIAL        |
*-------------------------------------------------------* ;

PATTERN V=M2;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF CANNS BY DAY';
TITLE2 F=TRIPLEX 'BY TRIAL';
FOOTNOTE 'AIRCRAFT TYPE <--> F4G';
PROC GCHART;
     BY TNUM;
     VBAR DAY / TYPE     = MEAN
                FRAME
                SUMVAR   = CANNAMT;
RUN;


*-------------------------------------------------------*
|         THIS SECTION WILL PRODUCE PIE CHARTS,         |
|        THE SLICES WILL CORRESPOND TO PART NUMBER      |
|        WITH EACH NEW PIE CORRESPONDING TO             |
|                   THE TRIAL NUMBER                    |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF CANNS BY PART/TRIAL';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE PART/SUMVAR  = CANNAMT
                PCT      = OUTSIDE;
RUN;
```

```
FILENAME IN85 '[GORNTOC.SAS]F4GTAPE85.DAT';
DATA TAPE85;
  INFILE IN85;
  INPUT TNUM $ 5-6 DAY $ 12-13 ACTYPE $ 20 BASENUM $ 27 NMCSHR 32-34;
RUN;

*-------------------------------------------------------*
|        THIS SECTION WILL PRODUCE BAR CHARTS,          |
|       EACH BAR WILL REPRESENT TOTAL NUMBER OF         |
|                 NMCS HOURS BY DAY                     |
|           NEW GRAPH PRODUCED FOR EACH TRIAL           |
*-------------------------------------------------------* ;

PATTERN V=M2;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF NMCS HOURS BY DAY';
TITLE2 F=TRIPLEX 'BY TRIAL';
FOOTNOTE 'AIRCRAFT TYPE <--> F4G';
PROC GCHART;
     VBAR DAY / TYPE     = MEAN
                DISCRETE
                GROUP = TNUM
                FRAME
                SUMVAR    = NMCSHR;
RUN;


*-------------------------------------------------------*
|        THIS SECTION WILL PRODUCE PIE CHARTS,          |
|          THE SLICES WILL CORRESPOND TO DAY            |
|         WITH EACH NEW PIE CORRESPONDING TO            |
|                 THE TRIAL NUMBER                      |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF NMCS HOURS BY DAY/TRIAL';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE DAY/SUMVAR  = NMCSHR
                DISCRETE
                PCT     = OUTSIDE;
RUN;
```

NUMBER OF NMCS HOURS BY DAY

BY TRIAL



AIRCRAFT TYPE <--> F4C

```
FILENAME IN86 '[GORNTOC.SAS]F4Gtape86.DAT';
DATA TAPE86;
  INFILE IN86;
  INPUT TNUM $ 5-6 DAY 12-13 ACTYPE $ 20 BASENUM $ 27
        NMCSAC 33-34;
RUN;

*-------------------------------------------------------*
|         THIS SECTION WILL PRODUCE BAR CHARTS,         |
|        EACH BAR WILL REPRESENT TOTAL NUMBER OF        |
|                  NMCS ACFT BY DAY                     |
|            NEW GRAPH PRODUCED FOR EACH TRIAL          |
*-------------------------------------------------------* ;
PROC SORT; BY TNUM DAY;
PATTERN V=M2;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF NMCS ACFT BY DAY';
TITLE2 F=TRIPLEX 'BY TRIAL';
FOOTNOTE 'AIRCRAFT TYPE <--> F4G';
PROC GCHART;
     VBAR DAY /  TYPE     = MEAN
                 FRAME
                 DISCRETE
                 SUMVAR   = NMCSAC
                 GROUP    = TNUM;
RUN;


*-------------------------------------------------------*
|         THIS SECTION WILL PRODUCE PIE CHARTS,         |
|           THE SLICES WILL CORRESPOND TO DAY           |
|          WITH EACH NEW PIE CORRESPONDING TO           |
|                  THE TRIAL NUMBER                     |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF NMCS ACFT BY DAY/TRIAL';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE DAY/SUMVAR  = NMCSAC
                DISCRETE
                PCT     = OUTSIDE;
RUN;
```

```
FILENAME IN87 '[GORNTOC.SAS]F4Gtape87.DAT';
DATA TAPE87;
  INFILE IN87;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 PERSON $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------------*
|       THIS SECTION WILL PRODUCE X-Y PLOTS             |
|             OF DELAY TIME BY PERSON                   |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY      |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ON-EQUIP DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'TIME IS IN MINUTES';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC SORT; BY TNUM DAY PERSON;
PROC GPLOT;
BY TNUM DAY;
       PLOT TIME * PERSON =1;
RUN;

*-------------------------------------------------------*
|     THIS SECTION WILL PRODUCE PIE CHARTS,             |
|   THE SLICES WILL CORRESPOND TO PERSONNEL TYPE        |
|    WITH EACH NEW PIE CORRESPONDING TO THE DAY         |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY PERSONNEL TYPE';
TITLE2 F=TRIPLEX 'ON-EQUIPMENT';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE PERSON/SUMVAR  = DELNUM
                PCT     = OUTSIDE;
RUN;
```

```
FILENAME IN88 '[GORNTOC.SAS]F4GTAPE88.DAT';
DATA TAPE88;
  INFILE IN88;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 PERSON $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-----------------------------------------------------*
|       THIS SECTION WILL PRODUCE X-Y PLOTS           |
|          OF DELAY TIME BY PERSONNEL TYPE            |
|   A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY   |
*-----------------------------------------------------* ;

PROC SORT; BY TNUM DAY PERSON;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'OFF-EQUIP DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'TIME IS IN MINUTES';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM DAY;
       PLOT TIME * PERSON =1;
RUN;

*-----------------------------------------------------*
|      THIS SECTION WILL PRODUCE PIE CHARTS,          |
|    THE SLICES WILL CORRESPOND TO PERSONNEL TYPE     |
|     WITH EACH NEW PIE CORRESPONDING TO THE DAY      |
*-----------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY PERSONNEL TYPE';
TITLE2 F=TRIPLEX 'OFF-EQUIPMENT';
FOOTNOTE1 F=NONE 'AIRCRAFT <--> F-16';
PROC GCHART;
BY TNUM;
        PIE PERSON/SUMVAR  = DELNUM
                PCT      = OUTSIDE;
RUN;
```

```
FILENAME IN89 '[GORNTOC.SAS]F4GTAPE89.DAT';
DATA TAPE89;
  INFILE IN89;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 AGE $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------------*
|       THIS SECTION WILL PRODUCE X-Y PLOTS             |
|             OF DELAY TIME BY AGE TYPE                 |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY      |
*-------------------------------------------------------* ;

PROC SORT; BY TNUM DAY AGE;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ON-EQUIP DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'CAUSED BY AGE TYPE';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM DAY;
       PLOT TIME * AGE =1;
RUN;

*-------------------------------------------------------*
|      THIS SECTION WILL PRODUCE PIE CHARTS,            |
|      THE SLICES WILL CORRESPOND TO AGE TYPE           |
|     WITH EACH NEW PIE CORRESPONDING TO THE DAY        |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY AGE TYPE';
TITLE2 F=TRIPLEX 'ON-EQUIPMENT';
FOOTNOTE1 F=COMPLEX 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE AGE/SUMVAR  = DELNUM
                PCT     = OUTSIDE;
RUN;
```

```
FILENAME IN90 '[GORNTOC.SAS]F4GTAPE90.DAT';
DATA TAPE90;
  INFILE IN90;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 AGE $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------------*
|       THIS SECTION WILL PRODUCE X-Y PLOTS             |
|              OF DELAY TIME BY AGE TYPE                |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY      |
*-------------------------------------------------------* ;

PROC SORT; BY TNUM DAY AGE;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'OFF-EQUIP DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'FOR AGE TYPE';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM DAY;
       PLOT TIME * AGE =1;
RUN;

*-------------------------------------------------------*
|      THIS SECTION WILL PRODUCE PIE CHARTS,            |
|      THE SLICES WILL CORRESPOND TO AGE TYPE           |
|     WITH EACH NEW PIE CORRESPONDING TO THE DAY        |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY AGE TYPE';
TITLE2 F=TRIPLEX 'OFF-EQUIPMENT';
FOOTNOTE1 F=COMPLEX 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE AGE/SUMVAR  = DELNUM
                PCT     = OUTSIDE;
RUN;
```

```
FILENAME IN91 '[GORNTOC.SAS]F4GTAPE91.DAT';
DATA TAPE91;
  INFILE IN91;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 PART $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-----------------------------------------------------*
|      THIS SECTION WILL PRODUCE X-Y PLOTS            |
|            OF DELAY TIME BY PART NO.                |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY    |
*-----------------------------------------------------* ;

PROC SORT; BY TNUM DAY PART;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'FOR PART NUMBER';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM DAY;
       PLOT TIME * PART =1;
RUN;

*-----------------------------------------------------*
|     THIS SECTION WILL PRODUCE PIE CHARTS,           |
|     THE SLICES WILL CORRESPOND TO PART NUMBER       |
|    WITH EACH NEW PIE CORRESPONDING TO THE DAY       |
*-----------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY AGE TYPE';
FOOTNOTE1 F=COMPLEX 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE PART/SUMVAR  = DELNUM
                PCT      = OUTSIDE;
RUN;
```

```
FILENAME IN92 '[GORNTOC.SAS]F4GTAPE92.DAT';
DATA TAPE92;
  INFILE IN92;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 SHOP $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-------------------------------------------------------*
|      THIS SECTION WILL PRODUCE X-Y PLOTS              |
|            OF DELAY TIME BY SHOP NO.                 |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY      |
*-------------------------------------------------------* ;

PROC SORT; BY TNUM DAY SHOP;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'ON-EQUIP DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'FOR SHOPS';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM DAY;
      PLOT TIME * SHOP =1;
RUN;

*-------------------------------------------------------*
|     THIS SECTION WILL PRODUCE PIE CHARTS,             |
|     THE SLICES WILL CORRESPOND TO AGE TYPE            |
|    WITH EACH NEW PIE CORRESPONDING TO THE DAY         |
*-------------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY SHOP NUMBER';
TITLE2 F=TRIPLEX 'ON-EQUIPMENT';
FOOTNOTE1 F=COMPLEX 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
       PIE SHOP/SUMVAR  = DELNUM
                PCT     = OUTSIDE;
RUN;
```

```
FILENAME IN93 '[GORNTOC.SAS]F4GTAPE93.DAT';
DATA TAPE93;
  INFILE IN93;
  INPUT TNUM $ 5-6 DAY $ 12-13 BASENUM $ 20 SHOP $ 25-27
        TIME 30-34 DELNUM 40-41;
RUN;

*-----------------------------------------------------*
|       THIS SECTION WILL PRODUCE X-Y PLOTS           |
|             OF DELAY TIME BY SHOP                   |
|  A NEW PLOT WILL BE DRAWN FOR EACH TRIAL AND DAY    |
*-----------------------------------------------------* ;

*-----------------------------------------------------*
|      THIS SECTION WILL PRODUCE PIE CHARTS,          |
|      THE SLICES WILL CORRESPOND TO SHOP NUMBER      |
|     WITH EACH NEW PIE CORRESPONDING TO THE DAY      |
*-----------------------------------------------------* ;

PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'NUMBER OF DELAY BY SHOP NUMBER';
TITLE2 F=TRIPLEX 'OFF-EQUIPMENT';
FOOTNOTE1 F=COMPLEX 'AIRCRAFT <--> F-4G';
PROC GCHART;
BY TNUM;
        PIE SHOP/SUMVAR  = DELNUM
                PCT      = OUTSIDE;
RUN;


PROC SORT; BY TNUM DAY SHOP;
PATTERN V=M5;
GOPTIONS NOTEXT82;
TITLE1 F=XSWISS 'OFF-EQUIP DELAY TIME BY DAY/TRIAL';
TITLE2 F=TRIPLEX 'FOR SHOPS';
FOOTNOTE1 F=ITALIC 'AIRCRAFT <--> F-4G';
SYMBOL1   W=1 I=JOIN V=STAR;
PROC GPLOT;
BY TNUM DAY;
       PLOT TIME * SHOP =1;
RUN;
```

END

8-87

DTIC